

VŠB – Technická univerzita Ostrava
Fakulta elektrotechniky a informatiky

BAKALÁŘSKÁ PRÁCE

2017

Patrik Koch

VŠB – Technická univerzita Ostrava
Fakulta elektrotechniky a informatiky
Katedra kybernetiky a biomedicínského inženýrství

Využití simulačních nástrojů pro podporu tvorby aplikací pro
programovatelné automaty
Simulation Tools in Design of Control Applications for Programmable
Controllers

Zadání bakalářské práce

Student:

Patrik Koch

Studijní program:

B2649 Elektrotechnika

Studijní obor:

2612R041 Řídicí a informační systémy

Téma:

Využití simulačních nástrojů pro podporu tvorby aplikací
pro programovatelné automaty
Simulation Tools in Design of Control Applications for Programmable
Controllers

Jazyk vypracování:

čeština

Zásady pro vypracování:

1. Rozbor simulačních přístupů využitelných při návrhu řídicích aplikací v průmyslové automatizaci.
2. Seznámení se s vybranými HW a SW nástroji pro simulaci systémů.
3. Návrh simulace laboratorního a procesního modelu využívaného v laboratoři programovatelných automatů.
4. Návrh a realizace řídicí aplikace pro uvedený model.
5. Ověření funkčnosti simulace a řídicí aplikace.
6. Zhodnocení dosažených výsledků.

Seznam doporučené odborné literatury:

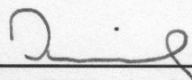
- [1] BERGER, Hans. *Automating with SIMATIC*. 5th edition. Erlangen, Germany: Publicis Publishing, 2013, 284 p. ISBN 978-3895783876.
- [2] BERGER, Hans. *Automating with SIMATIC S7-1500: Configuring, Programming and Testing with STEP 7 Professional*. Hardcover, 2014. ISBN-13: 978-3895784040.
- [3] Technická dokumentace k systému Simatic.

Formální náležitosti a rozsah bakalářské práce stanoví pokyny pro vypracování zveřejněné na webových stránkách fakulty.

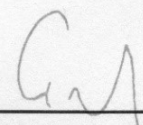
Vedoucí bakalářské práce: **doc. Ing. Jiří Koziorek, Ph.D.**

Datum zadání: 01.09.2016

Datum odevzdání: 28.04.2017


doc. Ing. Jiří Koziorek, Ph.D.
vedoucí katedry





prof. RNDr. Václav Snášel, CSc.
děkan fakulty

Prohlášení studenta

Prohlašuji, že jsem tuto bakalářskou práci vypracoval samostatně. Uvedl jsem všechny literární prameny a publikace, ze kterých jsem čerpal.

V Ostravě dne: 28. dubna 2017


.....
podpis studenta

Poděkování

Rád bych poděkoval panu doc. Ing. Jiřímu Koziorkovi, Ph.D. za odbornou pomoc a konzultaci při vytváření této bakalářské práce.

Abstrakt

Tato bakalářská práce se zabývá využitím simulačních nástrojů v průmyslové automatizaci. Hlavním úkolem bylo vytvořit simulační modely, včetně jejich řídicí aplikace, pro šest pracovišť nacházejících se v laboratoři programovatelných automatů. Všechny simulační modely byly zhotoveny pomocí simulačního nástroje SIMIT 9 ve verzi Ultimate. Řídicí aplikace byly vytvořeny v programovacím prostředí TIA Portal V13. Komunikace mezi PLC a simulačním modelem byla provedena s využitím průmyslového standartu OPC DA. Pro testování byla použita simulační technika Hardware-in-the-Loop. Testování proběhlo na všech simulačních modelech a případné nedostatky byly modifikovány. Řídicí aplikace byla testována i na laboratorním modelu, kde byla nutná drobná úprava kódu. Vytvořené modely dokáží poskytnout budoucím studentům určitý komfort v podobě rychlejšího pochopení systému a usnadnění vývoje aplikace pro tyto laboratorní pracoviště.

Klíčová slova

Simulace, Simulační nástroje, Simit, Programovatelný logický automat

Abstract

This bachelor thesis deals with the utilization of simulation tools in industrial automation. The main task was to create simulation models, including their control applications, for six workplaces in the laboratory of programmable automata. All simulation models were made by using SIMIT 9 simulation tool in version Ultimate. Control Applications were created in the TIA Portal V13 programming environment. Communication between the PLC and the simulation model was performed by using the industry standard OPC DA. Hardware-in-the-Loop technique was used for testing, Testing took place on all simulation models and possible deficiencies were modified. The control application was also tested on a laboratory model where minor code adjustment was required. The created models can provide future students with some comfort in the form of faster system understanding and ease of application development for these laboratory workplaces.

Key Words

Simulation, Simulation tools, Simit, Programmable logic controller

Obsah

Seznam použitých symbolů a zkratk	8
Seznam ilustrací	9
Seznam tabulek	11
Úvod.....	12
1. Simulace.....	13
1.1 Modely systému	13
1.1.1 Diskrétní a spojitý	13
1.1.2 Statický a dynamický	14
1.1.3 Deterministický a stochastický	14
1.2 Složky systému.....	15
1.3 Postup při návrhu simulačního modelu.....	16
1.4 Simulační techniky používané během vývoje řídicí aplikace	18
1.4.1 Model-in-the-Loop (MiL)	18
1.4.2 Software-in-the-Loop (SiL).....	19
1.4.3 Processor-in-the-Loop (PiL)	19
1.4.4 Hardware-in-the-Loop (HiL).....	19
1.5 Výhody a nevýhody	19
2. Simulace v průmyslové automatizaci.....	20
2.1 Používané simulační přístupy.....	21
2.1.1 Diskrétní simulace (DES).....	21
2.1.2 Systémová dynamika (SD).....	22
2.1.3 Agentově orientované modelování (ABM)	23
2.1.4 Metoda Monte Carlo (MCM)	23
2.1.5 Hybridní simulace	23
3. Simulační Nástroje	25
3.1 Visual Components 4.0	25
3.2 SIMIT 9	26
3.2.1 Integrace se softwarem COMOS Walkinside.....	29
3.3 AnyLogic.....	30
3.4 Devize	31
3.5 ReplicAnt VR.....	32

4.	Návrh simulačního modelu	34
4.1	Pracoviště distribuční	36
4.2	Pracoviště testovací	39
4.3	Pracoviště procesní	42
4.4	Pracoviště manipulační	45
4.5	Pracoviště třídící	48
4.6	Pracoviště mixovací	51
5.	Realizace řídicí aplikace	54
5.1	Pracoviště distribuční	55
5.2	Pracoviště testovací	56
5.3	Pracoviště procesní	57
5.4	Pracoviště manipulační	58
5.5	Pracoviště třídící	59
5.6	Pracoviště mixovací	59
6.	Ověření funkčnosti	61
6.1	Nastavení OPC serveru	61
6.2	Nastavení OPC klienta	64
6.3	Testování aplikace na simulačním modelu	65
6.4	Testování aplikace na laboratorním modelu	66
	Závěr	67
	Literatura	68
	Seznam příloh	72

Seznam použitých symbolů a zkratek

Zkratka	Význam
3D	Trojrozměrný
ABM	Agent-Based Model
ABS	Agent-Based Simulation
AGV	Automated Guided Vehicle
ANT	Applying New Technologies
CAD	Computer-Aided Design
CPU	Centrální procesorová jednotka
DEDS	Discrete Event Dynamic Simulation
DES	Discrete Event Simulation
DESS	Discrete Event Static Simulation
HiL	Hardware-in-the-Loop
HW	Hardware
ITS	Immersive Training Simulator
MCM	Monte Carlo Method
MCS	Monte Carlo Simulation
MiL	Model-in-the-Loop
OB	Organization Block
OPC DA	OPC Data Access
OPC UA	OPC Unified Architecture
PiL	Processor-in-the-Loop
PLC	Programmable Logic Controller
PLE	Personal Learning Editor
SCADA	Supervisory Control and Data Acquisition
SD	System Dynamics
SiL	Software-in-the-Loop
SW	Software
TON	Timer On-Delay
TP	Pulse Timer
VR	Virtuální Realita

Seznam ilustrací

Obrázek 1.1:	<i>Stavová proměnná diskrétního systému [2]</i>	14
Obrázek 1.2:	<i>Stavová proměnná spojitého systému [2]</i>	14
Obrázek 1.3:	<i>Složky systému [5]</i>	15
Obrázek 1.4:	<i>Kroky při návrhu simulačního modelu [1]</i>	16
Obrázek 1.5:	<i>V model [6]</i>	18
Obrázek 2.1:	<i>Simulace průmyslových robotů – svařování karoserie automobilu [13]</i>	20
Obrázek 2.2:	<i>Diskrétní simulace [16]</i>	21
Obrázek 2.3:	<i>Systémová dynamika [20]</i>	22
Obrázek 2.4:	<i>Agentově orientované modelování [21]</i>	23
Obrázek 3.1:	<i>Simulace průmyslových robotů v automobilovém průmyslu [27]</i>	26
Obrázek 3.2:	<i>Příklad diagramu s propojenými komponenty a řídicími prvky [32]</i>	27
Obrázek 3.3:	<i>Popis uživatelského rozhraní SIMIT [32]</i>	29
Obrázek 3.4:	<i>Prostředí COMOS Walkinside [35]</i>	30
Obrázek 3.5:	<i>Simulace provozu křižovatky [43]</i>	31
Obrázek 3.6:	<i>Simulace pevnosti v tahu ocelového autodílu [44]</i>	31
Obrázek 3.7:	<i>Simulátor využívající virtuální realitu [48]</i>	32
Obrázek 4.1:	<i>Nastavení Coupling v SIMIT před vývojem simulačního modelu</i>	34
Obrázek 4.2:	<i>Projektová navigace</i>	35
Obrázek 4.3:	<i>Seznam materiálů</i>	35
Obrázek 4.4:	<i>Laboratorní model distribučního pracoviště</i>	36
Obrázek 4.5:	<i>Simulační model distribučního pracoviště</i>	38
Obrázek 4.6:	<i>Laboratorní model testovacího pracoviště</i>	39
Obrázek 4.7:	<i>Simulační model testovacího pracoviště</i>	41
Obrázek 4.8:	<i>Laboratorní model procesního pracoviště</i>	42
Obrázek 4.9:	<i>Simulační model procesního pracoviště</i>	44
Obrázek 4.10:	<i>Laboratorní model manipulačního pracoviště</i>	45
Obrázek 4.11:	<i>Simulační model manipulačního pracoviště</i>	47
Obrázek 4.12:	<i>Laboratorní model třídícího pracoviště</i>	48
Obrázek 4.13:	<i>Simulační model třídícího pracoviště</i>	50
Obrázek 4.14:	<i>Laboratorní model mixovacího pracoviště</i>	51
Obrázek 4.15:	<i>Simulační model mixovacího pracoviště</i>	53
Obrázek 5.1:	<i>Hardware konfigurace PLC</i>	54
Obrázek 5.2:	<i>Návrh Petriho sítí pro distribuční pracoviště</i>	55
Obrázek 5.3:	<i>Návrh Petriho sítí pro testovací pracoviště</i>	56
Obrázek 5.4:	<i>Návrh Petriho sítí pro procesní pracoviště</i>	57
Obrázek 5.5:	<i>Návrh Petriho sítí pro manipulační pracoviště</i>	58
Obrázek 5.6:	<i>Návrh Petriho sítí pro třídící pracoviště</i>	59
Obrázek 5.7:	<i>Návrh Petriho sítí pro mixovací pracoviště</i>	60
Obrázek 6.1:	<i>Přidání nového kanálu v KEPServerEX</i>	61
Obrázek 6.2:	<i>Přidání nového zařízení v KEPServerEX</i>	62
Obrázek 6.3:	<i>Vkládání proměnných v KEPServerEX</i>	63
Obrázek 6.4:	<i>OPC Quick Client</i>	63

Obrázek 6.5:	<i>Nastavení OPC klienta v SIMIT</i>	64
Obrázek 6.6:	<i>Propojení PLC se SIMIT</i>	65
Obrázek 6.7:	<i>Hardware konfigurace modulu vzdálených vstupů/výstupů</i>	66
Obrázek 6.8:	<i>Propojení PLC s modulem vzdálených vstupů/výstupů</i>	66

Seznam tabulek

Tabulka 1.1:	<i>Příklady systémů a jeho částí [2]</i>	16
Tabulka 3.1:	<i>Typy řídicích prvků [32]</i>	27
Tabulka 3.2:	<i>Typy grafických prvků [32]</i>	28
Tabulka 4.1:	<i>Vstupní a výstupní signály distribučního pracoviště</i>	36
Tabulka 4.2:	<i>Vstupní a výstupní signály testovacího pracoviště</i>	39
Tabulka 4.3:	<i>Vstupní a výstupní signály procesního pracoviště</i>	43
Tabulka 4.4:	<i>Vstupní a výstupní signály manipulačního pracoviště</i>	46
Tabulka 4.5:	<i>Vstupní a výstupní signály třídícího pracoviště</i>	48
Tabulka 4.6:	<i>Vstupní a výstupní signály mixovacího pracoviště</i>	52
Tabulka 5.1:	<i>Hardware konfigurace PLC a adresy</i>	54
Tabulka 6.1:	<i>Hardware konfigurace modulu vzdálených vstupů/výstupů a adresy</i>	66

Úvod

Simulační nástroje jsou využívány již řadu let. V dnešní době, kdy jde vývoj neustále kupředu, je čím dál větší zájem o simulace za pomoci kterých, můžeme vyvíjet rozsáhle továrny a jejich řídicí systém ještě, než jsou vybudovány a uvedeny do provozu. Simulace je využívána také pro urychlení školení budoucích zaměstnanců.

Ze začátku bakalářské práce je čtenáři přiblížen obecný pojem simulace a je seznámen s typy modelu systému včetně jeho rozdělení. V kapitole byl také popsán podrobný postup, krok za krokem, vedoucí k správnému navržení simulačního modelu. Objevuje se zde také zmínka o V modelu, na kterém jsou rozvrženy používané simulační techniky.

Z informací získaných ze zahraniční literatury byly zjištěny a analyzovány nejpoužívanější simulační přístupy využívané v průmyslové automatizaci. Celkem bylo popsáno pět nejčastějších typu, z čehož jeden je kombinací několika různých přístupů. Ke každému přístupu byl poté popsán vybraný simulační nástroj.

Po seznámení s vybranými simulačními nástroji byl pro návrh modelu zvolen software SIMIT 9 vyvinutý společností Siemens, umožňující simulovat technické procesy v reálném čase. V práci bylo vytvořeno šest simulačních modelů odpovídajících jednotlivým pracovištím nacházejících se v laboratoři programovatelných automatů. Řídicí aplikace byly uskutečněny v programovacím jazyce kontaktních schémat s využitím softwaru TIA Portal V13, jehož součástí je vývojové prostředí Simatic Step 7.

Řídicí aplikace byly otestovány na vytvořených simulačních modelech pomocí průmyslového komunikačního standardu OPC DA. Pro testování byl využit programovatelný automat typu Simatic S7-1500 nacházející se v laboratoři. Aplikace byla také otestována na laboratorním modelu, kde musel být k PLC připojen modul vzdálených vstupů a výstupů umožňující vzdáleně řídit laboratorní model.

1. Simulace

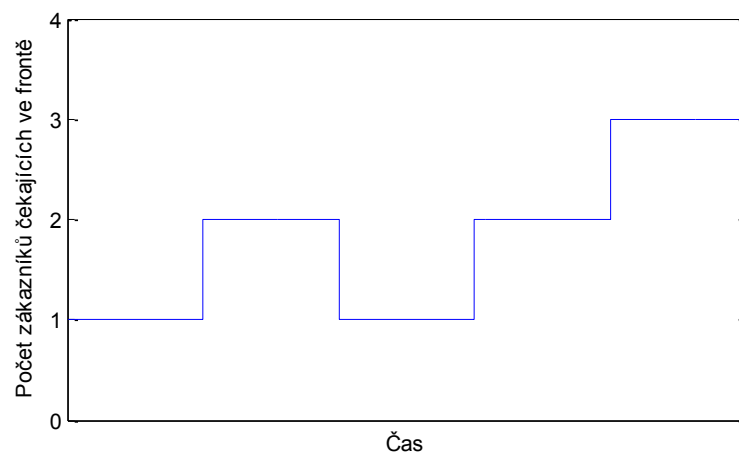
Simulace je v literatuře popisována jako napodobení skutečného systému nebo procesu pomocí modelu systému závislého na čase, prostřednictvím kterého můžeme nalézt vady a problémy vyskytující se v systému a přispět tak k jejich včasnému odstranění. Jakékoliv napodobování systému nebo procesu může být tedy nazýváno jako simulace. Cílem simulace je pomocí vytvořeného simulačního modelu zkoumat chování daného systému. Tento model má obvykle podobu souborů s předpoklady týkajícími se fungování systému. Tyto předpoklady jsou vyjádřeny v matematických, logických a symbolických vztazích mezi entitami nebo předměty zájmu systému. V případě, že je systém jednoduchý, jeho model může být popsán a vyřešen analyticky. [1; 2; 3]

1.1 Modely systému

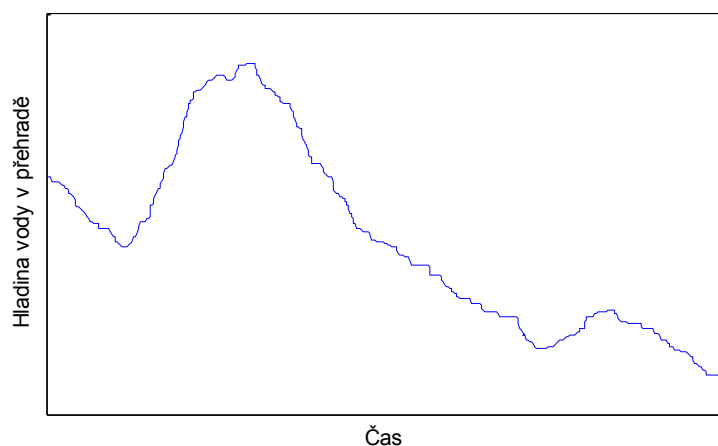
Modely jsou definovány jako vyobrazení systému pro účel jeho bližšího přezkoumání. Model systému by měl být dostatečně podrobně popsán, aby se co nejvíce podobal reálnému systému. Vytvoření modelu za účelem provádění experimentů se obecně označuje jako simulace. Můžeme se setkat s matematickými a fyzickými modely. Fyzický model vyobrazuje skutečný předmět v jiném velikostním měřítku. Matematický model používá symbolický zápis a matematické rovnice pro ztvárnění systému. Simulační model je typ matematického modelu a je dělen podle daných kritérií do několika skupin. [2; 4; 5]

1.1.1 Diskrétní a spojitý

Hlavním rozdělením modelu systému v oblasti simulace je dělení na diskrétní a spojitý. V diskrétním systému se hodnoty stavových proměnných mění v diskrétních časových okamžicích, kdežto ve spojitém systému se hodnoty stavových proměnných mění průběžně v čase. Stavová proměnná systému popisuje stav, který odráží hodnoty proměnných. Matematicky můžeme diskrétní model popsat diferenčními rovnicemi, zatímco spojitý model můžeme popsat diferenciálními rovnicemi. Příkladem diskrétního systému je banka, stavová proměnná, počet zákazníků v bance, se změní jen ve chvíli, když dorazí další zákazník nebo zákazníkovi byla poskytnuta žádaná služba. Příkladem spojitého systému je hladina vody v přehradní nádrži. Během nějakého času po bouři steče voda do nádrže. Voda je z hráze čerpána na ochranu proti povodním a pro výrobu elektřiny a také se odpařuje, čímž se snižuje vodní hladina. Stavová proměnná, hladina vody, se tedy mění spojitě s časem. [1; 2; 4]



Obrázek 1.1: *Stavová proměnná diskrétního systému [2]*



Obrázek 1.2: *Stavová proměnná spojitého systému [2]*

1.1.2 Statický a dynamický

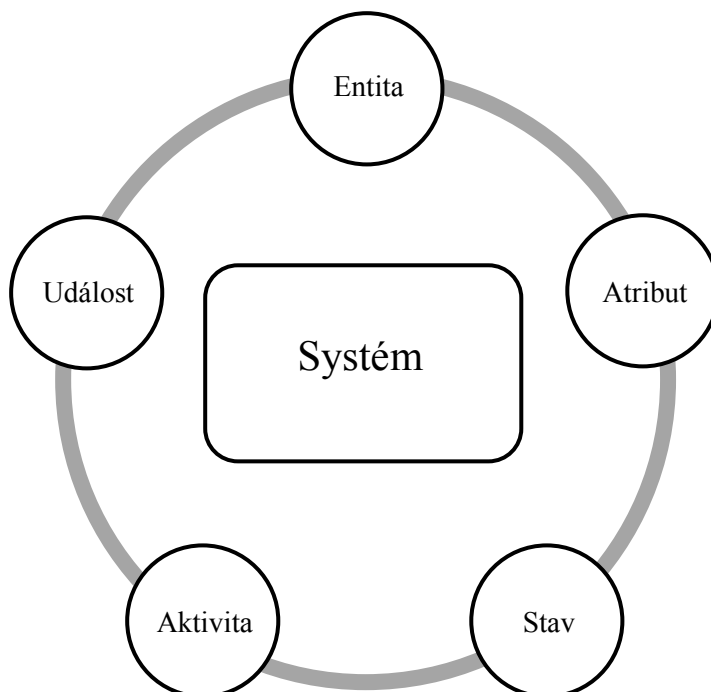
Druhé je rozdělení na statický a dynamický model. Statický model, někdy nazývaný jako simulační model Monte Carlo, představuje systém v konkrétním časovém okamžiku, zatímco dynamický model reprezentuje systém, jehož chování se mění s postupem času. Příkladem statického systému je hod mincí a dynamického systému je například simulace banky od 9:00, do 16:00. [1; 2]

1.1.3 Deterministický a stochastický

Model se také dělí na deterministický a stochastický. Deterministický model má deterministické proměnné, což znamená, že žádná proměnná neobsahuje náhodná čísla. Všechny vstupní hodnoty jsou známé. Deterministický příklad nastane, pokud všichni pacienti dorazí do zubní ordinace v plánovaném čase schůzky. Proměnné ve stochastickém modelu mají pravděpodobnostní charakter a to znamená, že jejich hodnoty jsou náhodné. Mají tedy jednu nebo více náhodných vstupních proměnných. Simulace banky by obvykle obsahovala náhodné časy příjezdu zákazníku a náhodné doby trvání služby

vykonávané zákazníkovi. Ve stochastické simulaci musí být výstup (průměrný počet lidí ve frontě, průměrný čas čekání zákazníka) považován za statistický odhad skutečných vlastností systému. [1; 2]

1.2 Složky systému



Obrázek 1.3: *Složky systému* [5]

Pro schopnost správně analyzovat systém je potřeba definovat několik jeho pojmů. Předmět zájmu v systému se nazývá entita. Atribut je proměnná entity. Aktivita udává časový úsek o určité délce. V případě, že analyzujeme např. banku, jednou z entit může být zákazník, stav jeho účtu může být atribut a vklad peněz na účet může být aktivita. [2]

Stav systému je definován jako soubor proměnných potřebných pro popis systému v libovolném čase ve vztahu k předmětům zájmu. U banky je to například počet zaneprázdněných pokladníků, počet zákazníků čekajících ve frontě nebo u přepážky a čas příjezdu dalšího zákazníka. [2]

Událost je definována jako okamžitý případ situace, kdy by mohlo dojít k změně stavu systému. Označení endogenní je používáno k popisu aktivit a událostí probíhajících v rámci systému a označení exogenní se používá k popisu aktivit a událostí, které ovlivňují systém z vnějšku. V příkladu s bankou je exogenní událost příjezd zákazníka a endogenní událost je ukončení prováděné služby zákazníkovi. [2; 5]

1. Simulace

Tabulka 1.1: *Příklady systémů a jeho částí* [2]

Systém	Entity	Atributy	Aktivity	Události	Stavové proměnné
Banka	Zákazníci	Kontrola stavu účtu	Vklad peněz	Příchod, odchod	Počet zaneprázdněných pokladníků, počet zákazníku ve frontě
Železnice	Strojvedoucí	Počáteční stanice, konečná destinace	Cestování	Příjezd do stanice	Počet strojvedoucích čekajících na nádraží, počet strojvedoucích na cestě do stanice
Výroba	Stroje	Rychlost, výrobní kapacita, četnost poruch	Sváření, lisování	Porucha	Stav stroje (zaneprázdněn, v nečinnosti, mimo provoz)
Zásobárna	Sklad	Kapacita	Odběr zboží	Poptávka	Stav zásob

1.3 Postup při návrhu simulačního modelu

Pro správné provedení simulace a vytvoření simulačního modelu, je nutné provést postupně tyto jednotlivé kroky.



Obrázek 1.4: *Kroky při návrhu simulačního modelu* [1]

1. Simulace

Krok 1: Pochopení systému

K provedení simulace musíme v první řadě porozumět systému. Jedná se o velice důležitý krok, na kterém závisí správné provedení celé simulace. Pokud systém neprostudujeme důkladně, výsledné proměnné mohou být chybné. Pochopením systému odhalíme jeho chování a následně můžeme určit, zda je vhodné použít diskrétní nebo spojitou simulaci. [1]

Krok 2: Stanovení cílů

V tomto kroku je nutné si stanovit všechny cíle. Jako cíle můžeme nazvat otázky, na které, za pomoci simulace, chceme znát odpověď. Zde se musíme i zamyslet, jestli je simulace vhodná metoda pro náš problém a podle toho se rozhodneme, jestli simulaci provést. [1; 2]

Krok 3: Sběr dat

Existuje souhra mezi konstrukcí modelu a získáváním potřebných vstupních dat. S rostoucí složitostí modelu systému, roste počet vstupních dat, která budeme potřebovat. Sběr dat zabere mnoho času a je proto důležité začít co nejdříve. [2]

Krok 4: Volba modelu

Volba modelu se odvíjí od konkrétní aplikace a závisí na jeho správném vytvoření. Musíme si promyslet, s jakým modelem budeme dále pracovat. V případě, že vše provedeme správně, vývoj modelu zabere rozumné množství času a vyhneme se spoustě programovacím hazardům. Na závěr si zvolíme software určený k modelování, který nejvíce vyhovuje naší aplikaci. [1]

Krok 5: Vytvoření modelu

Zde vytvoříme skutečný simulační model, včetně jeho prvního spuštění a otestování. Po vytvoření modelu můžeme přejít k dalšímu kroku. [1]

Krok 6: Kontrola vytvořeného modelu

Vytvořený model je překontrolován, zda skutečně vyobrazuje simulovaný systém a zda jsme schopni získat požadované výsledky. Pokud model nepředstavuje reálný systém, musíme zopakovat všechny předcházející kroky. Pro přechod do následujícího kroku by výsledky vyvinutého systému měli odpovídat výsledkům skutečného systému. [1]

Krok 7: Schválení výsledného modelu

Schválení je největší výzvou simulace. Je nutno zkontrolovat, jestli výsledky systému jsou shodné s výsledky reálného systému. V případě, že výsledky nesouhlasí, znamená to, že v jakémkoliv minulém kroku se nachází chyba. Postup návrhu musí být opakován od kroku 3, 4 nebo 5, podle závažnosti chyby. [1]

Krok 8: Experimentování

V této fázi musí být stanoveny alternativní situace, které mohou u reálného systému nastat. Experiment se opakuje několikrát, abychom se zbavili účinku nejistoty. [1; 2]

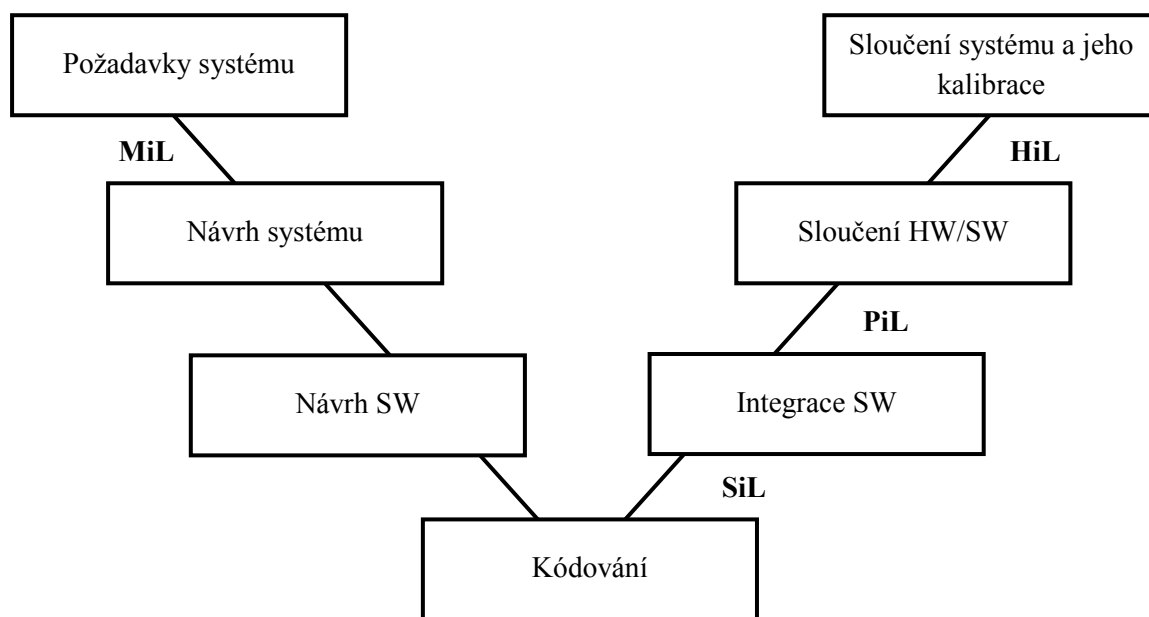
Krok 9: Spouštění a analýza simulačního modelu

V posledním kroku je provedeno spuštění hotového simulačního modelu. K analýze výsledků simulace jsou použity různé statistické nástroje. [1]

1.4 Simulační techniky používané během vývoje řídicí aplikace

Abychom ověřili správnost našeho postupu a mohli schválit splnění zadaných požadavků, můžeme využít X-in-the-Loop techniky k testování aplikace v jednotlivých fázích jejího vývoje. Kvůli stádiu ověřování a schvalování došlo k lehké úpravě tradičního vývojového procesu vodopád a byl vytvořen V model, který propojuje aktivity v počátečním vývoji s odpovídajícími aktivitami v pozdějším testování. V model se může objevit v několika variantách, podle potřeby vyhovující pro jeho využití. [4; 6]

V model je primitivní obměna tradičního vodopádového modelu systému nebo SW vývoje. V model má větší důraz na ověřování a schvalování. V model ohýbá směrem nahoru dolní polovinu vodopádového modelu, čímž vzniká tvar písmene V. Činnosti na pravé straně ověřují nebo schvalují činnosti nacházející se na levé straně. Levá strana zastupuje analýzu, která rozděluje požadavky na malé zvládnutelné kousky, zatímco pravá strana ukazuje syntézu, která slučuje kousky do systému, který odpovídá vyžadovaným potřebám uživatele. [6]



Obrázek 1.5: V model [6]

1.4.1 Model-in-the-Loop (MiL)

MiL je označení pro techniku, kde je model určité funkce kontrolován se simulovaným modelem systému. Simulace probíhá tak, že matematický model i řídicí systém jsou simulovány v simulačním prostředí na stejném počítači. Tato technika se používá k ověření a optimalizaci konstrukce stroje nebo technologie, k definování vstupních a výstupních signálů a pro návrh struktury řízení. MiL testy jsou prováděny za pomoci vývojového prostředí Matlab/Simulink. Testovací prostředí z MiL může být následně použito pro SiL testy. [4; 8]

1.4.2 Software-in-the-Loop (SiL)

SiL je ověřovací technika, kde je v simulačním (software) prostředí testován spustitelný kód, který může být vygenerován přímo z modelu. Je definován jako začlenění zkompilevaného software kódu do simulačního modelu. Matematický model je simulován v reálném čase společně s řídicím systémem na stejném počítači. SiL slouží k ověření výpočetní náročnosti řízení s danou periodou vzorkování a také pomáhá zvolit správný výkon řídicího hardwaru. [4; 9; 10]

1.4.3 Processor-in-the-Loop (PiL)

V technice PiL je procesor využíván pro testování spustitelného kódu. Matematický model je simulován v reálném čase. Řídicí systém je realizován na cílové HW platformě, procesor je připevněn na konkrétní hardware desce. Nejsou zde použity karty vstupů/výstupů, čidla a akční členy. Přenos dat mezi modelem a řídicím systémem je prováděn pomocí průmyslové komunikace. Využívá se pro ověření výpočetního výkonu řídicího HW, pro testování kritických situací atd. PiL vyplňuje mezeru mezi SiL a HiL. [4; 11]

1.4.4 Hardware-in-the-Loop (HiL)

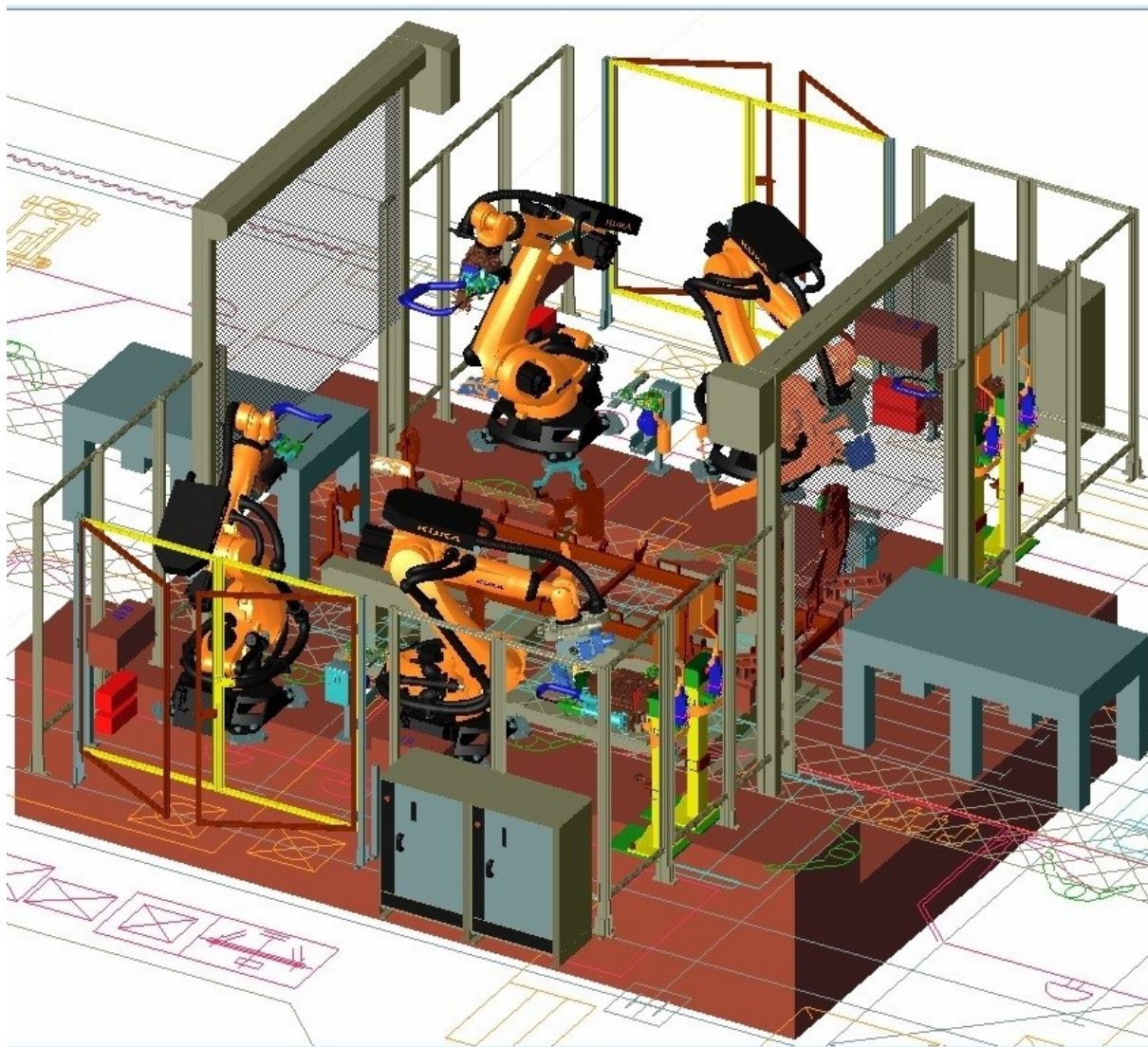
Pojem HiL označuje standardní metodu pro vývoj a testování řídicích aplikací před jejich uvedením do provozu. Části skutečného systému jsou nahrazeny simulací. Spustitelný kód je testován na cílovém hardware, zatímco jeho prostředí je simulováno. Reálné a simulované části systému jsou propojeny a uspořádány pro simulaci a testování dynamického chování skutečných komponent. HiL metoda je podobná jako PiL, simulace řídicího systému běží na cílové HW platformě. Navíc jsou použity karty vstupů/výstupů a model obsahuje emulátory senzorů a akčních členů. Jedná se o poslední krok před vývojem konečného prototypu. [4; 12]

1.5 Výhody a nevýhody

Simulace je modelování, díky kterému můžeme analyzovat rozsáhlé komplexní systémy. Největší výhodou simulace je, že ušetří spoustu peněz, protože zruší nutnost sledovat a zkoumat reálný systém před uvedením do provozu. Můžeme snadno analyzovat složité systémy. Simulace používá skutečná data, a proto jsou výsledky velmi blízko k realitě. Podmínky v simulaci mohou být lehce upravovány a výstupy tak mohou být jednoduše zkoumány. Kritické situace, kde by testování na reálném systému mohlo být dosti nákladné i nebezpečné, mohou být zkoumány bez rizika. Chování systému může být sledováno po delší dobu. Pro pečlivější prozkoumání systému může být měněna rychlost simulace. [1]

Mezi nevýhody patří vyšší počáteční náklady pro vytvoření simulačního modelu. Také může být nákladné měřit, jak jedna proměnná ovlivňuje druhou. Při vytváření simulace je nutno porozumět skutečnému fyzickému systému a také je zapotřebí znalosti o různých parametrech systému. [1]

2. Simulace v průmyslové automatizaci



Obrázek 2.1: Simulace průmyslových robotů – svařování karoserie automobilu [13]

Průmyslové systémy jsou jednou z nejvíce používaných oblastí využívající simulaci. Poprvé zde byla použita už v roce 1960. Simulace v průmyslu je úspěšně používána jako pomocník při návrhu nových výrobních zařízení, skladů a distribučních center. Využívá se také pro vyhodnocení navržených úprav stávajícího systému. Technici a analytici používáním simulace zjistili, že je podstatná pro posouzení celkových finančních investic do zařízení a výsledného návrhu zařízení. Manažeři zase zjistili, že je užitečná pro poskytnutí „zkušební jízdy“ před provedením investice, aniž by došlo k narušení stávajícího systému. [2; 14]

Abychom mohli analyzovat a optimalizovat výrobní systém, potřebujeme vědět způsob, jak vytvořit model a určit jeho provedení. Můžeme udělat mnoho různých typů modelů. Mnohdy je možné vytvořit hrubý odhad provedení továrny bez vyžadovaného modelování. V počátečních fázích vývoje procesu často nemůžeme udělat nic lepšího, než odhadnout jeho provedení, protože nemáme dost

podrobná data. Po nashromáždění více informací je možné továrny udělat s o něco přesnějšími výpočty. [1]

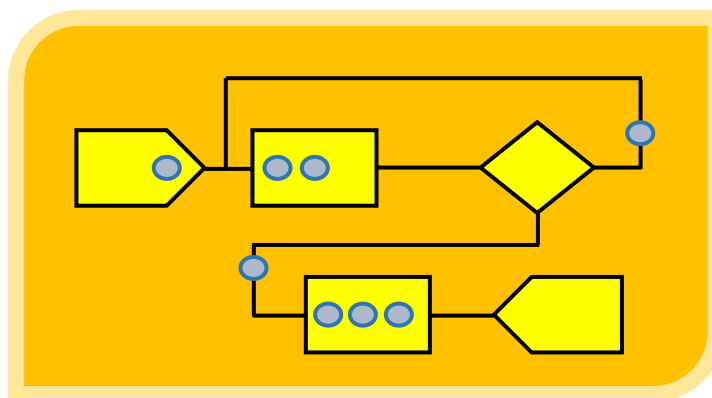
2.1 Používané simulační přístupy

Existuje spousta jednoúčelových simulátorů k simulaci velice specifických druhů systémů. Například pro simulaci pohybu vody a nečistot nebo vývoje galaxie. Klíčovou vlastností těchto nástrojů je, že jsou až moc specializované na vyřešení konkrétního problému. V mnoha případech tyto nástroje vyžadují velkou znalost k jejich použití. Jiné nástroje nejsou specializovány pro exaktní typ problému. Jsou to struktury všeobecného použití pro simulaci širšího rozsahu systémů. Dovolují uživateli modelovat, jak by se systém mohl chovat, nebo měnit s časem. Takové struktury lze chápat jako programovací jazyky vysoké-úrovně, umožňující uživateli simulovat mnoho různých druhů systému. [15]

Nejjednodušší a nejvíce používaný simulátor všeobecného použití je tabulkový procesor. I když jsou tabulkové procesory omezeny svou strukturou (představení komplexních dynamických procesů je složité, nedokážou zobrazit strukturu modelu graficky a pro zobrazení nejistoty se neobejdou bez speciálních doplňků), vzhledem ke své všudypřítomnosti jsou velmi široce použity pro prosté simulace, především v obchodním světě. [15]

Další nástroje všeobecného použití, které jsou lépe schopny vyobrazit komplexní dynamiku a také poskytnout grafické zobrazení struktury modelu (vlivový diagram nebo blokové schéma). Je finančně nákladnější a obtížnější se naučit pracovat s těmito nástroji než používat tabulky. Ve výsledku můžeme získat realistickou simulaci větších a složitějších systémů než za pomoci tabulek. Nástroje pro všeobecné použití jsou rozděleny do následujících kategorií. [15]

2.1.1 Diskrétní simulace (DES)



Obrázek 2.2: Diskrétní simulace [16]

Diskrétní simulace je využívána u více než 40 % případů použití simulace v průmyslu, jedná se tedy o nejpoužívanější techniku v této oblasti. V průmyslu je aplikována v celé řadě odvětví jako je například plánování a řízení výroby, technika procesu a management skladu. Zkráceně je označována jako simulační technika DES. Je vhodná pro podrobné, ale krátkodobější analýzy procesů. [17]

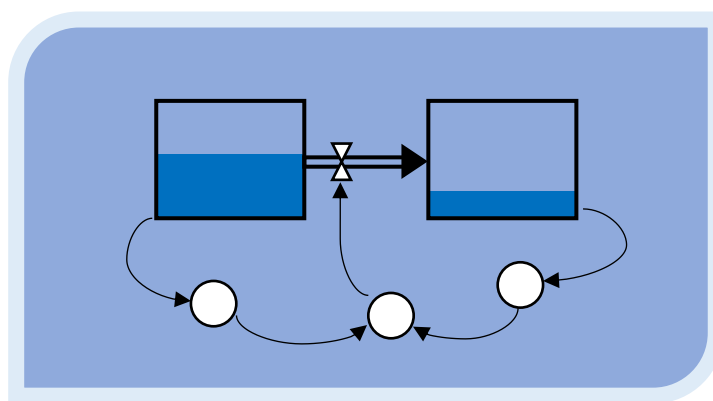
DES je poháněna změnou diskretních událostí, jejichž stav se mění v průběhu času. Modely se skládají z entit, zdrojů (prvky obsluhující entity) a řídicích prvků (prvky určující stavy entit a zdrojů).

DES není určena pro modelování pohybu spojitého systému nebo k vyobrazení spojitého systému pomocí diferenciálních rovnic. [15]

DES se dále rozlišuje na diskrétní statické systémy a diskrétní dynamické systémy. Základním rozdílem je, že v DESS se stav systému nevyvíjí v průběhu času. Simulace tohoto modelu zahrnuje opakované generování stavu systému a může být vykonáno s použitím algoritmů. U DEDS stav systému záleží na čase. Řazení do fronty je typickým příkladem DEDS. DESS obvykle zahrnuje odhad složitých vícerozměrných integrálů nebo součtu s pomocí simulační metody Monte Carlo. [18]

DES se zaměřuje na pozorovatelná a měřitelná hlediska výroby. Entity procházejí systémem lineárně. Neobjevují se zde zpětné vazby. DES model odráží podrobný rozbor chování systému. [19]

2.1.2 Systémová dynamika (SD)



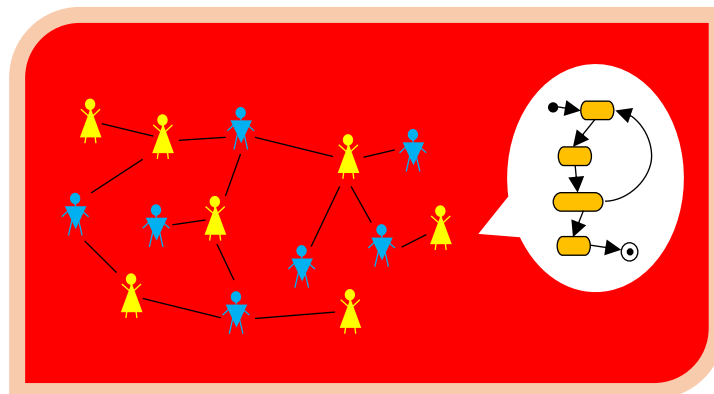
Obrázek 2.3: Systémová dynamika [20]

Simulační přístup systémová dynamika je druhá nejpoužívanější technika v průmyslu. Použita je u více než 15 % případů. Zaměřuje se především na oblasti jako je politika, strategický rozvoj, řízení projektů, řízení dodavatelského řetězce a analýza dynamických procesů, kde je nezbytné zachytit informační tok a zpětnou vazbu. SD byla kladně přijata v odvětvích jako automobilový průmysl, farmaceutický průmysl, výroba polovodičů, veřejné služby, pojišťovnictví, poradenství a vývoj software. [17; 19]

SD byla vyvinuta začátkem 60. let minulého století Jayem Forresterem v Massachusettském technologickém institutu. Jedná se o oblast, zaměřující se na chování systému, který můžeme popsat jako soubor vzájemně propojených částí, které neustále vzájemně reagují v průběhu času a vytváří tak jednotný celek. Dynamika se mění společně s časem. Systémová dynamika je proto metoda určená k porozumění, jak se systém mění v závislosti na čase. Zatímco se systém stává čím dál více složitější, jeho popis nemůže představovat všechny jeho části. Systémová dynamika se snaží uspořádat tuto složitou strukturu systémů. V struktuře by měli jít zřetelně vidět rozpory i řešení týkající se systému. [1; 19]

SD používá značný počet nástrojů, které představují chování modelu. Tyto modely pomáhají lépe pochopit systém. Některé častěji používané nástroje mohou pomoci uživateli ve vývoji chování modelu se systémovou dynamikou. [1]

2.1.3 Agentově orientované modelování (ABM)



Obrázek 2.4: Agentově orientované modelování [21]

Agentově orientované modelování, zkráceně označováno jako ABM nebo také ABS, je čtvrtý nejpopulárnější simulační přístup v průmyslu s použitím ve více než 5 % případů. Nejběžnější aplikace ABM se zaměřují na strategii, kde každá osoba v průmyslu je označována jako agent a každé chování agenta je modelováno se vztahem k strategickým konceptům. [17]

ABM je výpočetní technika pro modelování akcí a interakcí autonomních agentů. Cílem je posoudit efekt těchto agentů na systém jako celek (neposuzovat efekt každého agenta individuálně). ABM je vhodný pro modelování situací při kterých nejsou následky jasně zřetelné. ABM je schopno generovat komplexní proměnné vznikající z interakcí mezi agenty, a přitom jsou pravidla chování jednotlivých agentů jednoduchá. [22]

Agenti v systému jsou charakterizováni jako nezávislé, autonomní, samo se rozhodující složky. Můžeme se setkat s různými typy charakteristik agentů, které je vytváří flexibilní a přizpůsobivé dynamickému prostředí. [1]

2.1.4 Metoda Monte Carlo (MCM)

Simulační metoda Monte Carlo, zkráceně označována jako MCM nebo také MCS, je jedna z prvních vyvinutých simulačních technik. Poprvé byla představena Enricem Fermi během studie rozptylu neutronu v roce 1930. Označení Monte Carlo bylo poprvé použito matematikem Johnem Von Neumannem během druhé světové války v 40. letech minulého století. Jedná se o metodu pracující se stochastickým simulačním modelem, v simulaci se tedy objevují náhodně vygenerována čísla. Její použití v průmyslu je značně omezeno na řešení numerických problémů se stochastickými proměnnými, jako je například oceňování majetku a management rizik. [1; 17]

Princip simulace Monte Carlo lze vysvětlit na příkladu člověka házejícího dvěma šestistrannými kostkami. Jaká je pravděpodobnost, aby jejich součet byl 6 (1+5, 4+2, 3+3, atd.)? Jaká je šance na hod kombinace 3+3. Výsledek můžeme zjistit tak, že budeme kostkami házet mnohokrát, nejlépe milionkrát. Simulace MCM nám nabízí odhalení tohoto výsledku efektivně a rychle. [23]

2.1.5 Hybridní simulace

Až u 10 % případů v průmyslu je jako třetí nepoužívanější technika Hybridní simulace. Jedná se o kombinaci alespoň dvou simulačních přístupů. Nejznámějším příkladem je kombinace přístupů

DES a SD. Pomocí DES je hodnocen dopad na produkci a je zkoumána výkonnost podniku. SD zase zachycuje dlouhodobější účinky hodící se pro vyšší úroveň řízení. [17; 24]

Hybridní simulační model je používán v případech, kdy chceme současně využít spojitý a diskrétní prvky vyskytující se ve skutečném světě nebo pokud chceme sledovat chování modelu v makro a mikro úrovních, které se navzájem ovlivňují. Při vytváření hybridního modelu je potřeba definovat tři důležité prvky. Moduly, každý modul může používat jinou simulační metodu, propojení mezi nimi, včetně nahromadění a roztřídění informací a aktualizací pravidla popisující, jak informace zaslána modulem ovlivní ostatní moduly. [24]

3. Simulační Nástroje

Většina společností pracujících se simulačním nástrojem pro simulaci průmyslových aplikací, využívá spíše komerční software na místo programovacích jazyků, jako je například jazyk C. Simulační software je nejčastěji vybírán podle 2 kritérií, kterými jsou flexibilita modelování (schopnost vytvořit jakýkoliv model systému bez ohledu na jeho složitost nebo jedinečnost) a jednoduchost použití. [14]

V průmyslu existuje několik hlavních typů simulačního softwaru. Simulační jazyk je softwarový balíček, kde je model vyvíjen programováním. Tradiční programování znamená vývoj simulačního modelu psaním kódu, ale to už je minulostí. Dnešní simulační jazyky umožňují grafické vytváření modelu. Příkladem simulačních jazyků je Arena, AweSim, Extend, GPSS/H, Micro Saint, MODSIM III, SES/workbench, SIMPLE++, SIMSCRIPT II.5, SIMUL8 a SLX. Největší výhodou simulačních jazyků je flexibilita modelování a jejich nevýhodou je potřeba odborné znalosti programování. [14]

Výrobně-orientovaný simulační jazyk je ten, kde je modelování specializováno na výrobu nebo manipulaci s materiálem. Příkladem je software AutoMod a Quest. Výhodou je snížení doby programování, díky hotovým konstrukcím jako jsou pásy nebo AGV (automaticky řízené vozidlo, nejčastěji převážející materiál v továrně nebo ve skladu). [14; 25]

Protože byl značný zájem o to mít simulační software, který je jednoduchý pro vytvoření modelu, vznikl výrobně-orientovaný simulátor, což je simulační balíček navržený pro modelování výrobních systémů ve specifické třídě systému. Příkladem simulátorů je FACTOR/AIM, ProModel, Taylor II a WITNESS. Simulační model je vyvíjen graficky za pomoci simulátoru metodou „drag and drop“. Vybrané položky z menu jsou myší přetaženy do dialogových oken. Výhodou simulátoru je značné snížení doby potřebné na vývoj modelu. Nevýhodou je, že nejsou flexibilní jako simulační jazyky, protože nedovolují plnohodnotné programování. [14]

3.1 Visual Components 4.0

Společnost Visual Components byla založena v roce 1999 finskými a americkými experty na simulaci. Stejnomený 3D simulační software využívá přístupů diskrétní simulace. Software byl do roku 2014 dostupný v pěti samostatných verzích nabízející různé sady funkcí (3DAutomate, 3DCreate, 3DRealize, 3DRealize R a 3DSimulate). Nejnovější verze softwaru je nazývána Visual Components 4.0 a je dostupná ve třech možných verzích lišící se v množství obsahu (Essentials, Professional a Premium). [26; 27]

Essentials je hlavním produktem Visual Components 4.0. Umožňuje rychle vytvoření modelu pouhým přetažením komponent z knihovny a jejich následné propojení. Do 3D světa je možné importovat CAD soubory vytvořené v programech jako Autodesk, Dassault, PTC, Siemens atd. Produkt také podporuje propojení s reálným PLC s použitím průmyslového standardu OPC UA, pomocí čehož můžeme vizualizovat výrobu, sbírat a analyzovat informace, testovat a inovovat aplikaci před uvedením do provozu. [28]

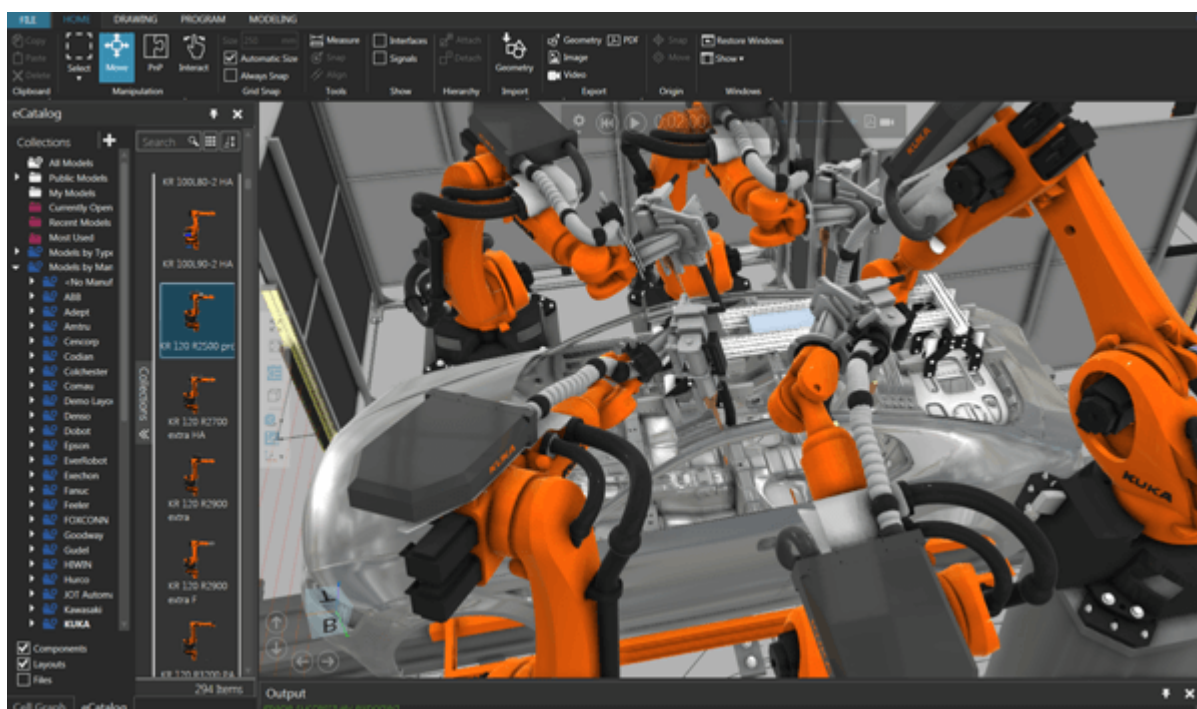
Produkt Professional obsahuje všechny funkce, jako produkt Essentials. Navíc obsahuje funkce jako je modelování komponent, která umožňuje vytvořit pohyblivé části z importovaných CAD

3. Simulační Nástroje

souborů, wizardy pro efektivní a jednoduché vytvoření komponent, snížení velikosti souborů a zvýšení výkonu simulace. [29]

Premium zahrnuje funkce obou předchozích produktů včetně dalších funkcí, jako je vizualizace lakování a následné měření tloušťky vrstvy, což je využitelné při procesu výroby automobilu. Mezi další dodatečné funkce patří nástroj pro automatizované plánování trasy robota na základě analýzy objektů a rychlé naučení pohybu robotů do konkrétních pozic, vhodné pro robotické svařování. Nasměrování strojů, lakování, operace s vodní tryskou, těsnění spojů a řezání. [30]

Knihovna komponent obsahuje přes 1200 3D modelů jako jsou stroje, dopravníky, podavače, nástroje, výrobní a stavení objekty, komponenty pro balení zboží, nespočet různých doplňků a samozřejmě robotů od více než 30 výrobců jako je ABB, Fanuc, KUKA, Mitsubishi, Panasonic atd. [26; 28]



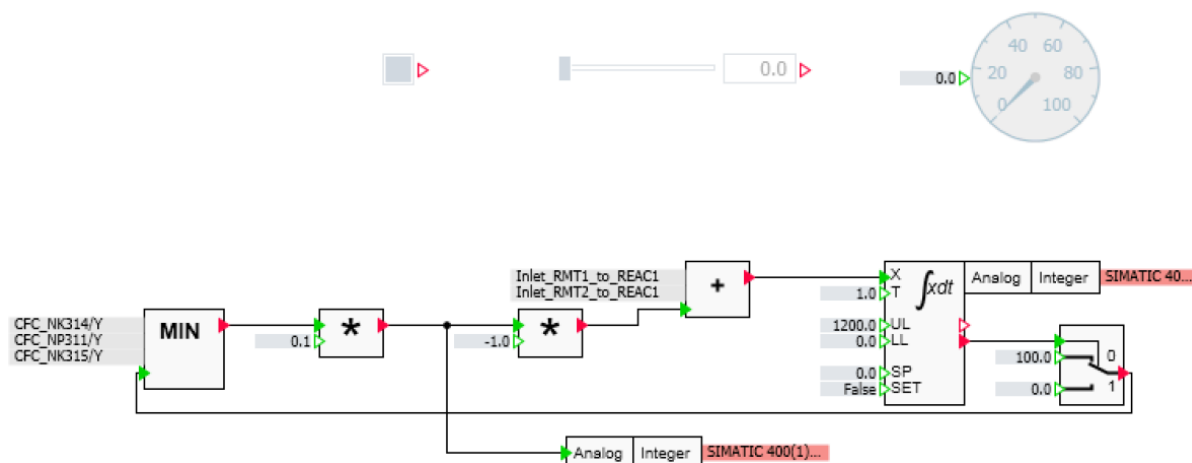
Obrázek 3.1: Simulace průmyslových robotů v automobilovém průmyslu [27]

3.2 SIMIT 9

SIMIT je simulační software vyvinutý společností Siemens. Jedná se o chytrou kombinaci testování a výcviku automatizačních projektů. Software vykonává simulaci v reálném čase a využívá přístupu systémové dynamiky, dynamika je v tomto případě čas. Uživatel může virtuálně spustit automatizační program, stroje a procesy. Prostředí je také využíváno pro realistický nácvik pro budoucí operátory továrny. Platformu SIMIT můžeme jednoduše používat bez dostupného vybavení nebo rozsáhlé technické znalosti modelování procesů. SIMIT 9 nabízí moderní a uživatelsky přívětivé grafické rozhraní pro intuitivní ovládání. Automatizační projekty mohou být realizovány včas, efektivně a spolehlivě. [31]

3. Simulační Nástroje

SIMIT může být použit jako simulátor pro testování vstupních a výstupních signálů PLC nebo jako simulátor továrny. Program nabízí mnoho možností pro testování a uvedení do provozu. I když z počátku použijeme SIMIT jen jako uživatelské rozhraní pro testování signálů, simulační modely můžeme přidat kdykoliv později. To nám dovoluje simulovat chování továrny a provádět dynamické testy. [32]











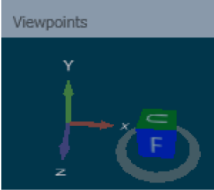


Obrázek 3.2: Příklad diagramu s propojenými komponenty a řídicími prvky [32]

Pro práci s nástrojem SIMIT je důležité znát následující elementy. *Diagram* obsahuje simulační model. Komponenty, řídicí prvky a propojení mezi nimi je graficky znázorněno. Diagramy se vytváří a upravují v editoru diagramu. Simulační model může obsahovat více diagramů. *Komponenty* poskytují logické a aritmetické funkce, pohony, senzory, propojení mezi nimi a komunikaci. *Řídicí prvky* slouží pro vkládání a zobrazování hodnot. *Vizualizace* umožňuje vytvářet diagramy s řídicími a zobrazovacími prvky, buďto manuálním vytvořením diagramů nebo automatickým vygenerováním diagramů z existujících dat. Grafické nástroje jako jsou čáry, obdélníky, elipsy atd. mohou být statické nebo animované, což umožňuje vytvářet komplexní dvourozměrné animace. Vizualizace poskytuje ideální přehled všech signálů v systému. Hodnoty, patřící do totožné části systému, mohou být rovněž uvedeny seskupeným způsobem, nezávisle na používaných fyzických adresách. *Propojení* definuje rozhraní, přes které bude SIMIT komunikovat s automatizovaným systémem. Mohou zde být importovány a upravovány signály. [32]








Tabulka 3.1: Typy řídicích prvků [32]

Název	Datový typ	Použití	Symbol
Binární displej	Binární	Zobrazení signálu	False
Analogový displej	Analogový, integer	Zobrazení signálu	
Digitální displej	Analogový, integer	Zobrazení signálu	0.0
Sloupcový graf	Analogový, integer	Zobrazení signálu	

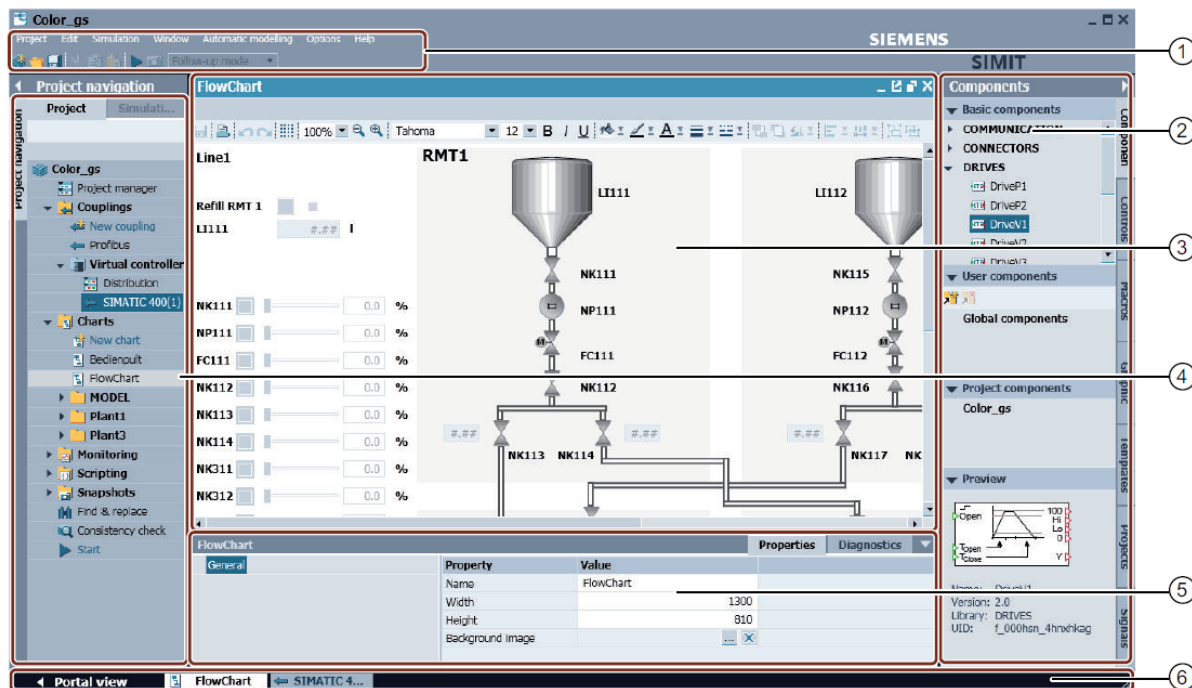
3. Simulační Nástroje

Tlačítko	Binární	Vstupní signál	
Tlačítko s obrázkem	Binární	Vstupní signál	
Přepínač	Binární	Vstupní signál	
Přepínač s obrázkem	Binární	Vstupní signál	
Stupňový přepínač	Integer	Vstupní signál	
Stupňový přepínač s obrázkem	Integer	Vstupní signál	
Digitální vstup	Analogový, integer	Vstupní signál	
Posuvník	Analogový	Vstupní signál	
3D prohlížeč	-	Integrace 3D grafiky do grafu	
Dělič signálu	-	Rozdělení signálů, musí být připojeny k jinému řídicímu prvku	
Akce	-	Otevírá grafy a trendy	

Tabulka 3.2: Typy grafických prvků [32]

Grafický prvek	Popis	Nastavení
 Text	Vloží text kdekoliv v diagramu	Písmo, velikost písma, barva písma, barva pozadí, barva ohraničení
 Line	Nakreslí rovnou čáru	Barva čáry, tloušťka čáry
 Rectangle	Nakreslí obdelník	Barva ohraničení, tloušťka ohraničení, barva výplně
 Ellipsis	Nakreslí elipsu	Barva ohraničení, tloušťka ohraničení, barva výplně
 Polyline	Nakreslí lomenou čáru	Barva čáry, tloušťka čáry
 Ellipse arc	Nakreslí oblouk elipsy	Barva čáry, tloušťka čáry
 Bezier curve	Nakreslí Bezierovu křivku	Barva čáry, tloušťka čáry

Pomocí SIMIT 9 je testování a optimalizace prováděna ve zcela virtuálním prostředí. To umožňuje spolehlivě a efektivně testovat všechny příslušné automatizační funkce ještě před uvedením do provozu. Téměř všechny nedostatky mohou být detekovány a opraveny před vlastním provozem. Přijetím stávajících plánovacích a technických dat SIMIT zajišťuje rychlejší a úspornější uvedení do provozu. [31]



Obrázek 3.3: Popis uživatelského rozhraní SIMIT [32]

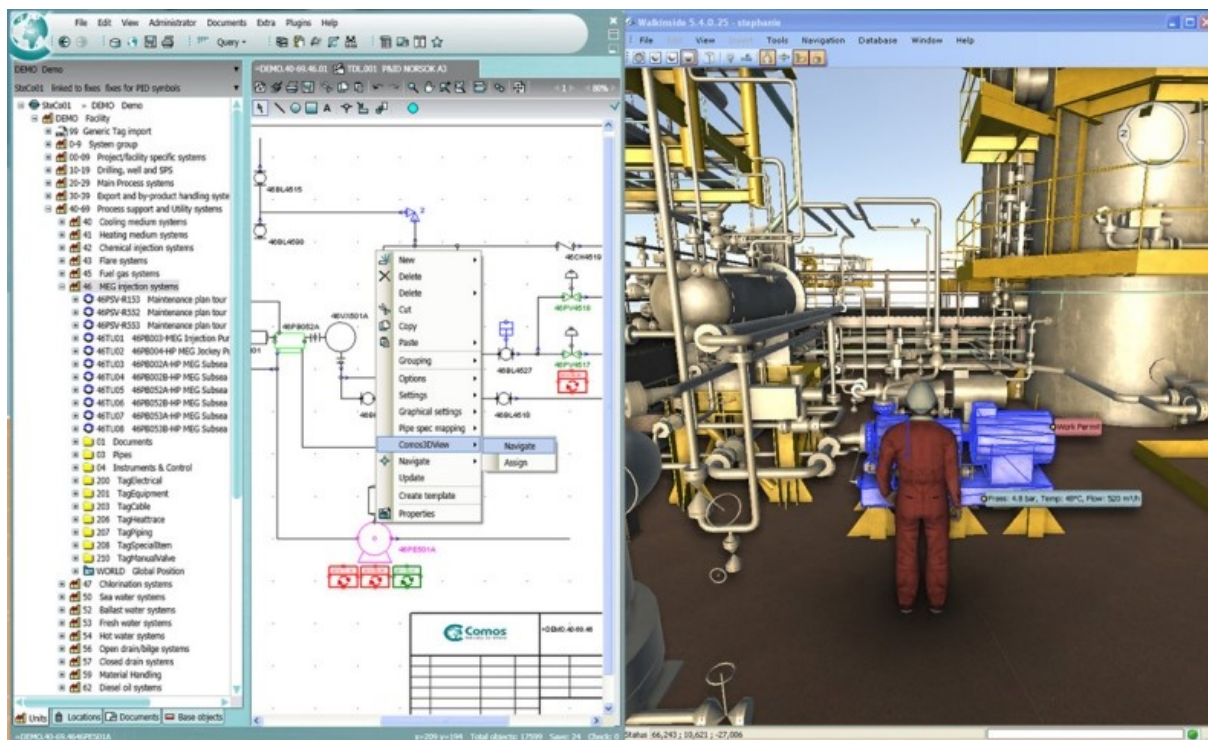
① Menu lišta a panel nástrojů. Přístup k funkcím SIMIT. ② Karty úloh. Objekty jako knihovna komponent, řídicí a grafické objekty. ③ Pracovní plocha. Místo, kde probíhá editování. ④ Navigace v projektu. Zobrazení aktuálního projektu ve stromové struktuře. ⑤ Pohled na proměnnou. Jsou zde zobrazeny proměnné vybraného objektu. ⑥ Zde lze přepínat mezi otevřenými editory a vzhledem portálu. Podle potřeby je zde zobrazena informace o aktuálním stavu SIMIT. [32]

3.2.1 Integrace se softwarem COMOS Walkinside

COMOS Walkinside umožňuje využití 3D technických dat ze základních a detailních fází vývoje. Software poskytuje bezproblémovou výměnu dat mezi simulační platformou SIMIT, čímž posouvá simulaci na vyšší úroveň. Vysoce komplexní modely továren mohou být vyobrazeny realisticky ve třech rozměrech. Továrny mohou být použity nejen pro strojírenské a monitorovací účely, ale také pro výcvik nových zaměstnanců. Servis a údržbářské práce mohou být efektivně plánovány a simulovány. [33; 34]

COMOS Walkinside se vyskytuje ve čtyřech verzích (Builder, Viewer, ITS a Integration). Produkt Builder automaticky vytváří hotové 3D virtuální modely největších průmyslových procesů továren. Viewer je výkonná 3D vizualizace aplikace v reálném čase, která vykresluje obrovské, složité, počítačem vygenerované modely. ITS ponoří operátory a techniky do továrny, kde simuluje scénáře, s kterými se mohou setkat ve skutečném životě. Podporuje více avatarů a je vhodná pro testování týmového výkonu a komunikace. [35; 36; 37]

3. Simulační Nástroje



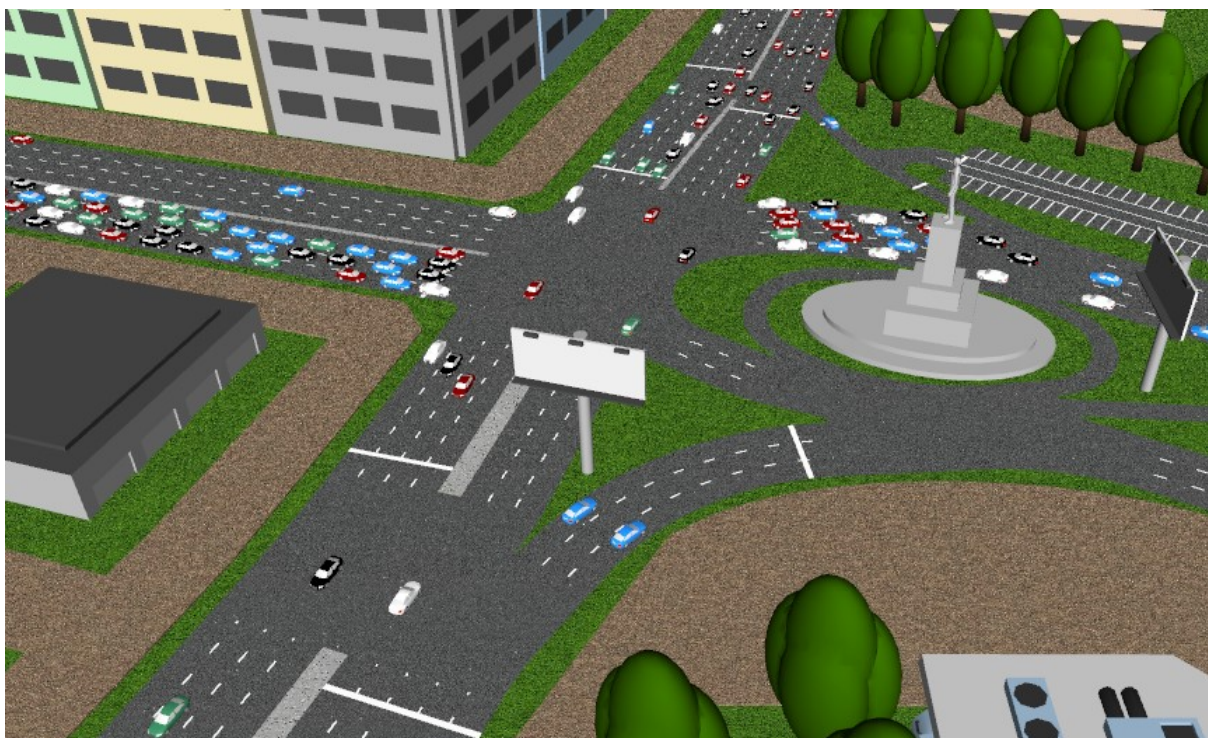
Obrázek 3.4: Prostředí COMOS Walkinside [35]

3.3 AnyLogic

Společnost AnyLogic je poskytovatelem dynamických simulačních nástrojů, technologií a poradenských služeb pro business aplikace. Software AnyLogic využívá hybridního simulačního přístupu. Jako jediný nástroj sdružuje systémovou dynamiku, diskrétní simulaci a agentově orientované modelování s použitím jednoho modelovacího jazyku a jednoho vývojového prostředí. [38]

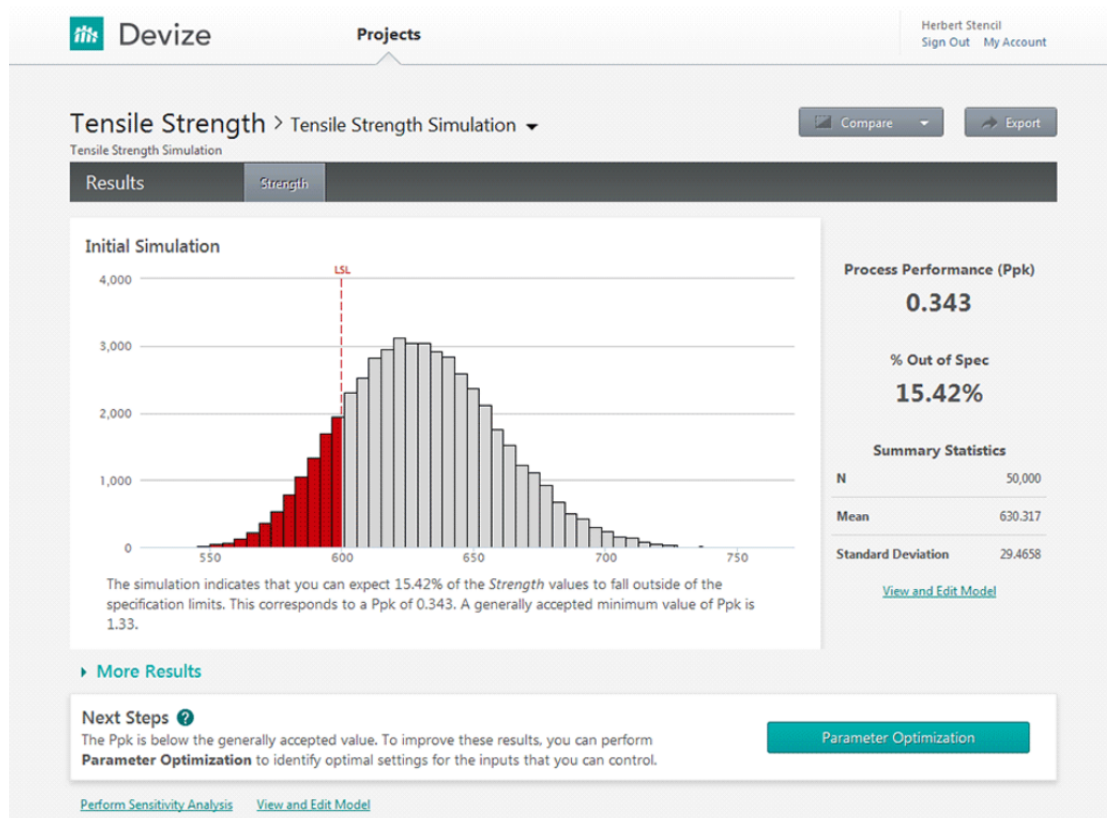
Flexibilní modelovací jazyk umožňuje zachytit složitost a různorodost obchodních, ekonomických a společenských systémů. Grafické rozhraní, nástroje a objekty z knihovny napomáhají k rychlému vývoji modelu v oblastech, jako je výroba, logistika, zdravotnictví, obchodní procesy, informatika a telekomunikace, doprava, proudění chodců atd. Prostředí poskytuje neomezené možnosti rozšíření pomocí Java kódu, externích knihoven a externích datových zdrojů. Dokončený model systému můžeme exportovat za pomoci jediného kliku myši, a tím vygenerovat spustitelnou Java aplikaci. [39; 40]

AnyLogic je nabízen ve třech edicích (PLE, Researcher a Professional). PLE je základní verze softwaru, která je k dispozici zdarma. Jejím účelem je sebevzdělávání a je určena především ke zkoumání světa simulace. Verze Researcher, už ze svého názvu vypovídá své určení. Je využívána především pro provádění veřejného výzkumu na univerzitě financovaného z dotací a grantů, její licenci lze zakoupit pouze jménem vzdělávací instituce. Obsahuje větší obsah a neomezené funkce jako bezplatná verze PLE. Pro soukromé výzkumy financované komerčními organizacemi a pro pracovníky v poradenských společnostech nebo vládních organizacích je určena verze Professional, která obsahuje všechny nabízené funkce pro plnohodnotné využívání softwaru. [41; 42]



Obrázek 3.5: Simulace provozu křižovatky [43]

3.4 Devize



Obrázek 3.6: Simulace pevnosti v tahu ocelového autodílu [44]

Devize je jednoduše použitelná webová aplikace s promyšleným designem vytvořená firmou Minitab, která poskytuje produkty a služby pro zlepšení kvality a statistiky vzdělání. Jedná se o program využívající simulační přístup Monte Carlo. Devize funguje na principu řízeného pracovního postupu. Jasně je určeno, jaké informace je třeba zadat a kdy. Vytváří se diagram, představující vztah mezi vstupy a výstupy, čímž můžeme ověřit, že je model přesný. Devize zobrazí klíčové výsledky srozumitelně, pomůže je interpretovat a identifikuje další kroky k vylepšení procesu. Program obsahuje vestavěnou srovnávací funkci a schopnost exportovat přímo do aplikace PowerPoint, díky čemuž je sdílení výsledku jednoduché. [45; 46; 47]

3.5 ReplicAnt VR



Obrázek 3.7: Simulátor využívající virtuální realitu [48]

ReplicAnt VR je software vyvinutý společností ANT Automation, která poskytuje technické služby, 3D modelování, vývoj a programování pro bezproblémové použití ReplicAnt. Je určený pro simulaci procesů, 3D vizualizaci a výcvik operátorů za pomoci virtuální a smíšené reality. Software je užitečný v mnoha různých fázích projektu, jako je návrh továrny, elektrárny, technologické linky, montáž, uvádění do provozu a inovace hotových procesů. Jeho využití spadá pod obory jako ocelářství, železářství, petrochemický průmysl, plynárenství, automobilový průmysl, výrobní průmysl a spousty jiných. [49]

S využitím fyziky a základů mechaniky program simuluje reálné události existující nebo nové fabriky. Systém dokáže, za pomoci rozsáhlé knihovny nástrojů, senzorů přítomnosti, fotosenzorů, detektorů kovu, bezdotykových senzorů, enkodérů, koncových spínačů a akčních členů včetně hydraulických pístů, motorů a dalších, simulovat vybrané vstupy a výstupy z PLC. [49]

ReplicAnt můžeme použít jako 3D SCADA systém pro vyobrazení procesů za pomoci 3D fotorealistického vykreslování v reálném čase a získání zpětné vazby továrny komunikující s PLC.

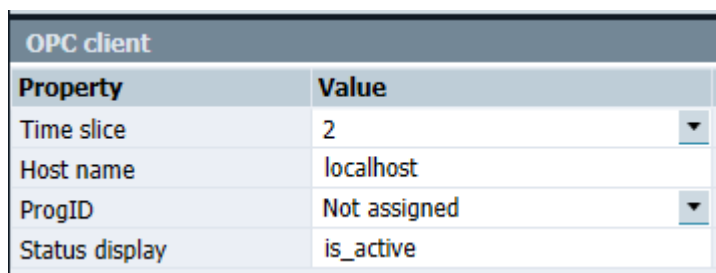
3. Simulační Nástroje

S pomocí ReplicAnt jsme také schopni provádět výcvik operátoru ve virtuálním světě, kde mohou pracovat s virtuálními panely a tlačítky uvnitř 3D modelu, nebo používat skutečná tlačítka připojena k PLC. Bezpečnostní školení může být provedeno ve virtuálním světě s pomocí simulování nebezpečných situací bez jakéhokoliv ohrožení zdraví účastníků. [49]

4. Návrh simulačního modelu

Po přezkoumání a pochopení příslušných laboratorních pracovišť a konzultaci s vedoucím práce byl pro návrh simulačních modelů vybrán software Simit od společnosti Siemens. Program umožňuje jednoduché vytváření simulačních modelů diskrétních i dynamických typů simulace. Modely jsou vytvořeny ve verzi Ultimate a nelze je otevřít ve verzi Demo, protože tyto verze mezi sebou nejsou kompatibilní. Všechny vytvořené projekty lze otevřít pouze s hardware klíčem. Projekty vytvořené ve verzi Demo lze otevřít jen v PC, kde byly vytvořeny a nelze je dále přenášet, proto byly modely vytvořeny v plné verzi softwaru.

Do každého projektu je vhodné ze začátku přidat propojení SIMIT s PLC výběrem volby „Add coupling“, kde je zvolen způsob komunikace s PLC. V případě vývoje modelu, kdy je problém připojit se k reálnému PLC nebo softwaru PLCSIM, například v kanceláři, je výhodné použít takové propojení umožňující spuštění simulace bez potřeby aktuálního připojení. Signály lze potom triviálně simulovat a provádět test modelu již během jeho vývoje. Například u volby „OPC client“ stačí jen na dolní liště zvolit u proměnné „ProgID“ možnost „Not assigned“, což dovoluje bez problému spustit simulaci. Žádné signály tedy nebudou přicházet do SIMIT.

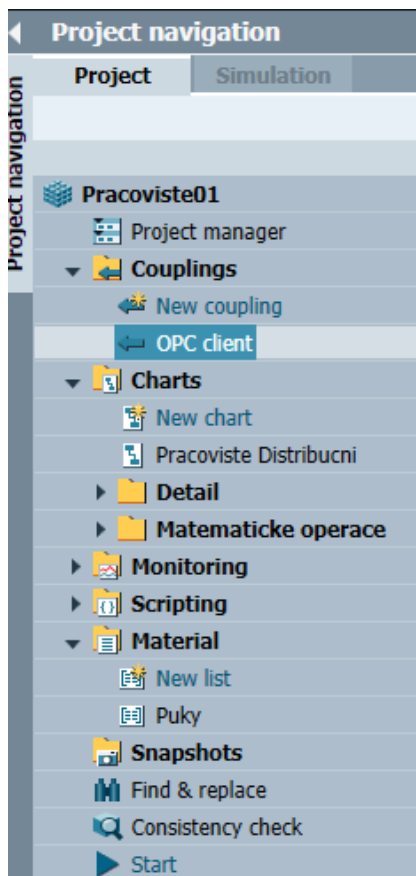


OPC client	
Property	Value
Time slice	2
Host name	localhost
ProgID	Not assigned
Status display	is_active

Obrázek 4.1: *Nastavení Coupling v SIMIT před vývojem simulačního modelu*

V levém panelu projektové navigace je zobrazeno již vytvořené propojení s OPC client ve složce „Couplings“, kde je možné zadat i další způsob propojení. Ve složce „Chart“ jsou umístěny modely, které jde rozdělit do několika podsložek. V tomto případě je u každého pracoviště umístěn hlavní diagram v kořenové složce, kde se nachází kompletní simulační model včetně jednoduché vizualizace. Dvě podsložky zase obsahují matematické operace, nezbytné pro správnou funkčnost celého modelu a detailní zobrazení jednotlivých pohonů nebo jiných zajímavých částí systému. Rozdělení diagramů je uživatelský libovolné, důležitá je především přehlednost.

V případech, kdy je pro simulaci použito dopravníkových pásů nelze pracovat bez potřebného materiálu. Ten je nutné zadat do seznamu materiálu, v tomto případě je pojmenován „Puky“.



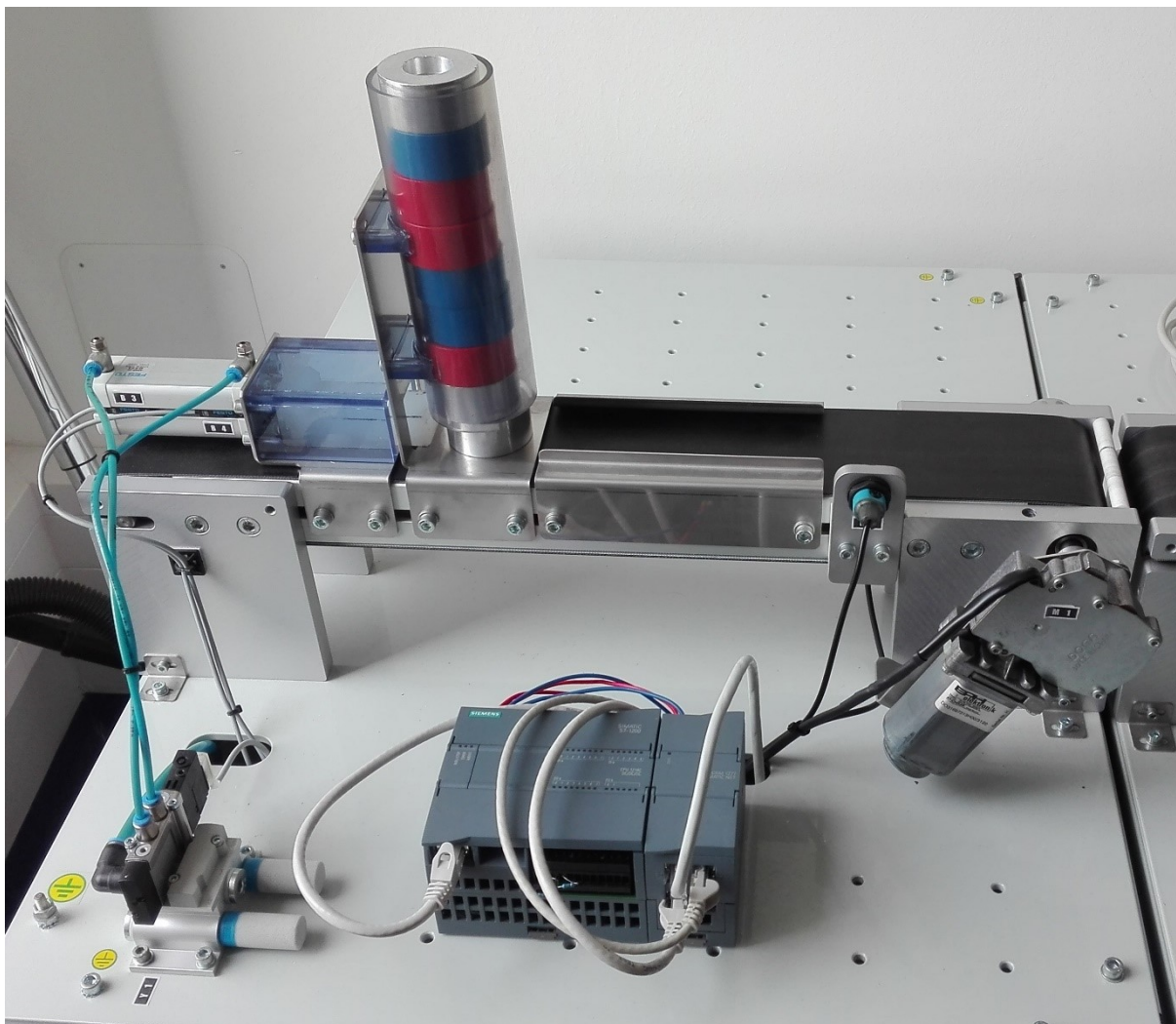
Obrázek 4.2: Projektová navigace

Z pravého panelu komponent stačí najít v knihovně CONTEC-MATERIAL prvek „CBox“, což je barevná krabice, a jednoduše jí pomocí metody „táhni a pusť“ vložit do seznamu materiálu. V tomto případě proběhlo vložení 10 prvků a u každého prvku byl nastaven název, šířka, výška a pomocí 3 vstupních parametru v dolní liště i barva. Pro vytváření simulačních modelů postačí 4 různé barevné kombinace materiálu. V modelech se měl původně přesouvat předmět válcového tvaru, pro tyto případy je tvar objektu ale nepodstatný.

Puky*						Components	
Reset filter						Components	Projects
Name	Width	Height	Depth	Type			
Cerveny1	500	500	0	CBox		Basic components	
Cerveny2	500	500	0	CBox		COMMUNICATION	
Cerveny3	500	500	0	CBox		CONNECTORS	
Modry1	500	500	0	CBox		CONTEC	
Modry2	500	500	0	CBox		CONVEYOR	
Modry3	500	500	0	CBox		MATERIAL	
Stribrny1	500	500	0	CBox		Box	
Stribrny2	500	500	0	CBox		CBox	
Stribrny3	500	500	0	CBox		CBoxDS256	
Zeleny	500	500	0	CBox		CVehicle	
						Vehicle	
						VehicleDS256	
						User components	

Obrázek 4.3: Seznam materiálů

4.1 Pracoviště distribuční



Obrázek 4.4: *Laboratorní model distribučního pracoviště*

Úkolem tohoto pracoviště je sekvenčně vyřádit jednotlivé puky umístěné v zásobníku pneumaticky poháněným podavačem na dopravníkový pás. Podavač puků obsahuje snímače koncových poloh. Pomocí elektrického motoru pohánějího pás je puk přemístěn na konec pracoviště, kde se nachází optický snímač zaznamenávající přítomnost puku. Snímač rotace pásu posílá jednotlivé impulzy do PLC v případě, že je pás v pohybu.

Tabulka 4.1: *Vstupní a výstupní signály distribučního pracoviště*

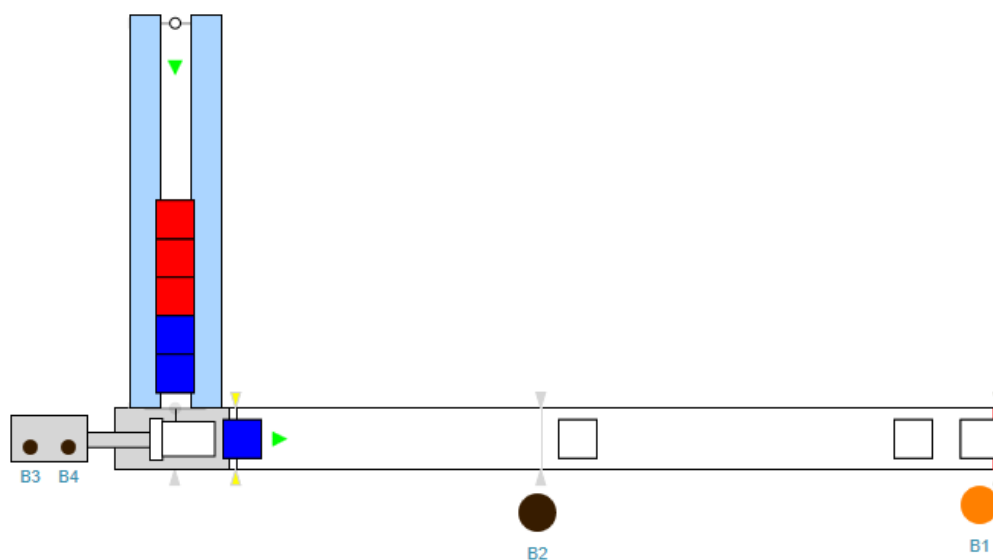
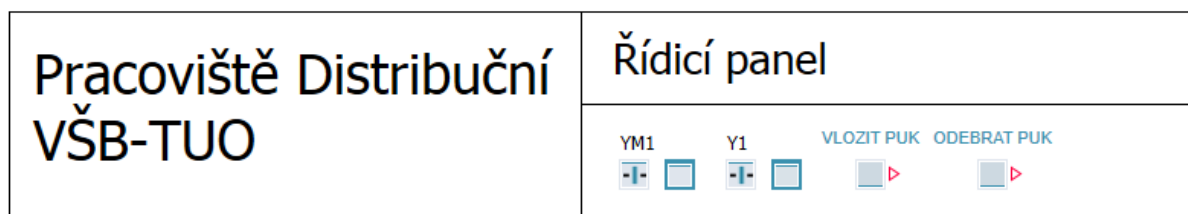
Vstupy			
Symbolický název	Datový typ	Adresa	Komentář
B1	Bool	%I0.0	Indukční snímač rotace pásu
B2	Bool	%I0.1	Snímač přítomnosti puku
B3	Bool	%I0.2	Podavač zasunut
B4	Bool	%I0.3	Podavač vysunut

4. Návrh simulačního modelu

Výstupy			
Symbolický název	Datový typ	Adresa	Komentář
Y1	Bool	%Q0.0	Aktivace podavače puku
YM1	Bool	%Q0.1	Motor dopravního pásu

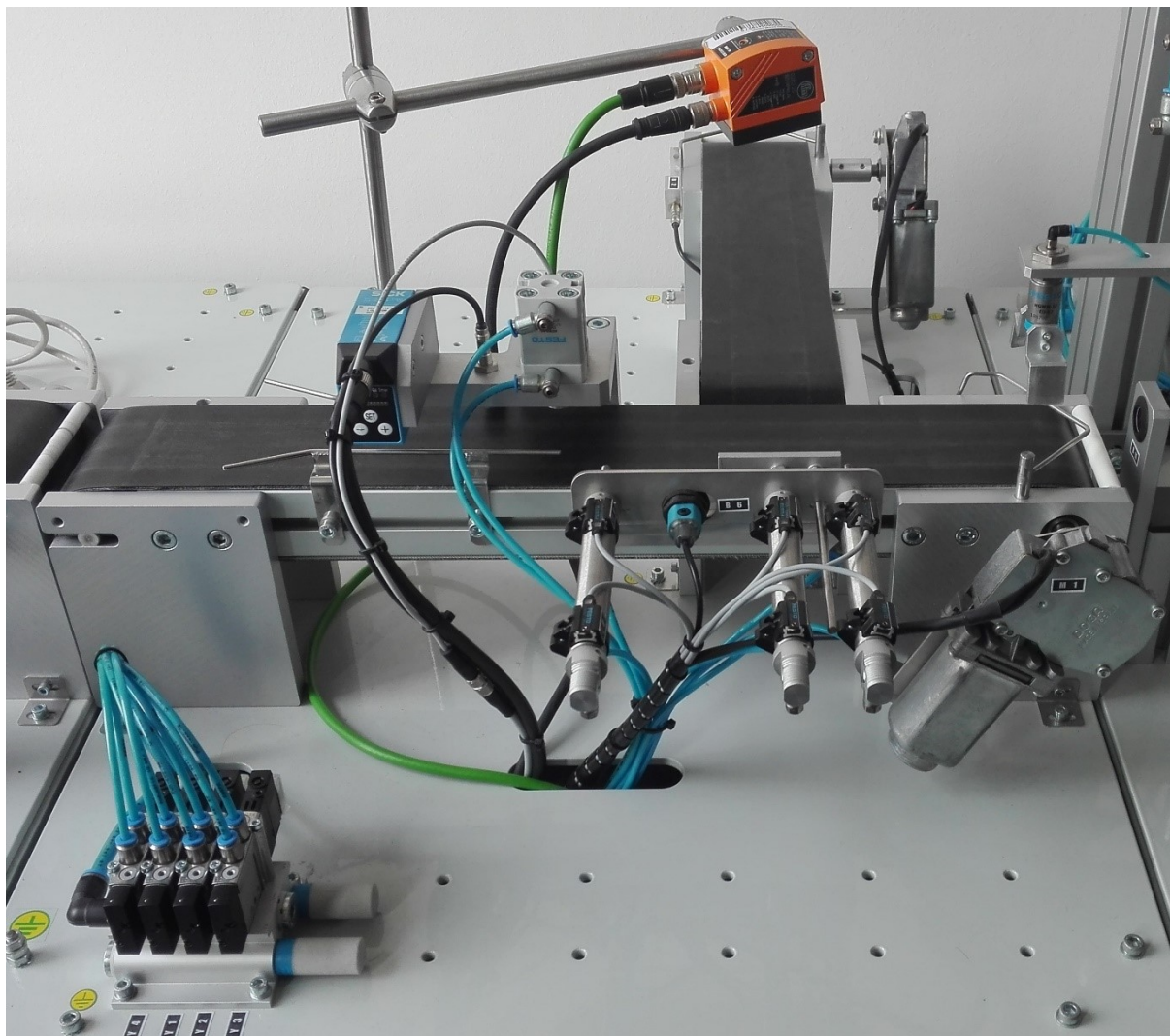
Simulační model je ovládán přes hlavní diagram pomocí řídicího panelu. Pro ovládání jednotlivých kontrolních prvků je nutné aktivovat prvek představující dělič signálu, který umožní ovládat přepínač nacházející se hned vedle. V případě, že jej ponecháme v počáteční poloze, budou výstupní signály řízeny z PLC a nebudeme je moct ovládat ručně. Pro vkládání puku do zásobníku stačí jednoduše kliknout na tlačítko „VLOZIT PUK“ a pro odebrání puku na konci pásu, je potřeba tlačítko „ODEBRAT PUK“ přidršet, dokud puk nezmizí z pásu.

Simulace přesunu materiálu a zásobník na puky jsou vytvořeny za pomoci knihovny „CONTEC“, ve které se nachází komponenty plnící různé funkce dopravníkového pásu. Motor pásu je simulován ve složce pro matematické operace, kde se nachází komponenta z knihovny „DRIVES“ usnadňující simulaci pohonů. Výstupem z komponenty „DriveP1“ je získána aktuální rychlost pásu, která řídí sériově řazené dopravníky v modelu. Integrací rychlosti je dráha, která nám postačí v rozsahu 0-360, což představuje úhel natočení motoru. Se znalostí pozice pootočení a využitím komponenty interval z knihovny „STANDARD“ je nasimulován snímač rotace pásu. Za jednu otáčku motoru vyšle snímač 8 pulzů, pro detailnější vyobrazení funkčnosti se ve složce projektu „Detail“ nachází podrobnější diagram. Pás představující zásobník má rychlost nastavenou na 100 %, jelikož puky jsou přitahovány gravitační silou. U dopravníkové komponenty zásobníku je nastaven počáteční počet puků na 9, stejně jako u laboratorního modelu. Také je nutné v dodatečných parametrech zapsat typ materiálu, v tomto případě „CBox“, jinak by se materiál po spuštění nezobrazil. Pod zásobníkem, ve vrstvě pod šedým grafickým objektem, se nachází skrytý pás, který posune materiál v případě, že podavač vyjíždí, což představuje vyražecí sílu. Při vyjíždění podavače, je na konci zásobníku aktivován „stopper“, aby nedošlo k průchodu puku skrz podavač. Simulace pohybu podavače je provedena pomocí komponenty „DriveP1“. Výstupem je získána aktuální pozice podavače, která je pak graficky vizualizována přes grafické nástroje za pomoci animace. Snímač přítomnosti je jednoduše nastaven na zvolenou pozici v parametrech komponenty pásu.



Obrázek 4.5: Simulační model distribučního pracoviště

4.2 Pracoviště testovací



Obrázek 4.6: *Laboratorní model testovacího pracoviště*

Testovací pracoviště přesune puky na konec prvního nebo druhého pásu podle získaných informací o puku v sekci měření na začátku pracoviště. Laboratorní model se skládá z dvou dopravníkových pásů, pro přemísťování puku, pneumaticky řízených závor, pro zadržení puku a pneumatického vyrážedce sloužícího k přesunu puku na druhý pás, včetně snímačů koncových poloh. Dále se zde nachází snímač barev pro detekci barvy puku, magnetický snímač a senzor výšky, který po vysunutí pístu zaznamená po přibližně 1 vteřině výšku puku v rozsahu 0-10 V, optický snímač přítomnosti před vyrážedcem a dva inkrementální snímače dopravníkového pásu.

Tabulka 4.2: *Vstupní a výstupní signály testovacího pracoviště*

Vstupy			
Symbolický název	Datový typ	Adresa	Komentář
B1	Bool	%I0.0	Indukční snímač rotace pásu č. 1
B2	Bool	%I0.1	Indukční snímač rotace pásu č. 2

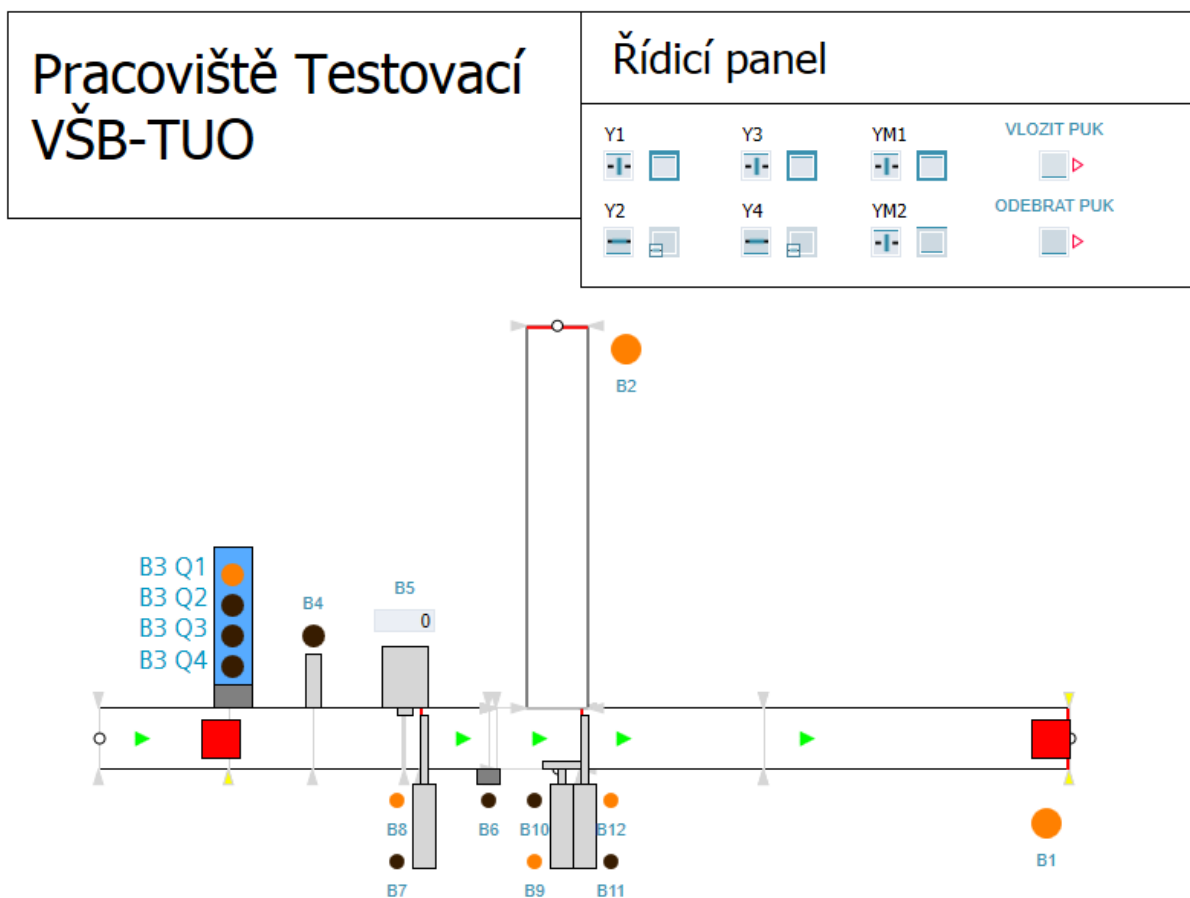
4. Návrh simulačního modelu

B3 Q1	Bool	%I0.2	Snímač barev - červená
B3 Q2	Bool	%I0.3	Snímač barev - modrá
B3 Q3	Bool	%I0.4	Snímač barev - stříbrná
B3 Q4	Bool	%I0.5	Snímač barev - nedefinovaná
B4	Bool	%I0.6	Snímač magnetických vlastností
B5	Int	%IW2	Senzor výšky
B6	Bool	%I0.7	Optický snímač přítomnosti na pase
B7	Bool	%I1.0	Závora 1 zasunuta
B8	Bool	%I1.1	Závora 1 vysunuta
B9	Bool	%I1.2	Vyrážeč zasunut
B10	Bool	%I1.3	Vyrážeč vysunut
B11	Bool	%I1.4	Závora 2 zasunuta
B12	Bool	%I1.5	Závora 2 vysunuta
Výstupy			
Symbolický název	Datový typ	Adresa	Komentář
Y1	Bool	%Q0.0	Výstup aktivující závoru 1
Y2	Bool	%Q0.1	Výstup aktivující vyrážeč
Y3	Bool	%Q0.2	Výstup aktivující závoru 2
Y4	Bool	%Q0.3	Výstup aktivující píst pro měření výšky
YM1	Bool	%Q0.4	Výstup aktivující elektromotor dopravníkového pásu č.1
YM2	Bool	%Q0.5	Výstup aktivující elektromotor dopravníkového pásu č.2

Dopravníkový pás je složen pomocí dvou různých typů komponent z knihovny „CONTEC“. Jedná se o komponenty pro zadržení puku a pro 90 stupňový přesun puku na druhý dopravník. Pro vkládání puku na začátek pásu je potřeba nastavit u prvního pásu v dodatečných parametrech typ materiálu „CBox“ a zapsat signál do vstupních proměnných pro vytvoření objektu. Rychlost horizontálně řazených dopravníků je dána výstupem komponenty „DriveP1“ pro motor č.1. Výstup komponenty „DriveP1“ pro motor č.2 řídí rychlost vertikálně natočeného pásu. Integrací individuálních rychlostí motoru získáme pozici pootočení v rozsahu 0-360, tak jako u simulačního modelu distribučního pracoviště. Pro první dopravníkovou komponentu je použito tří senzorů na různých pozicích pro snímač barvy, magnetických vlastností a výšky puku. U snímače barvy pás vrací integer hodnotu ID materiálu po přejetí dané pozice senzoru. V matematických operacích jsou nastaveny pomocí komponenty „interval“ vstupy do PLC. Magnetický snímač je v log. 1 u každého lichého ID puku. Snímač výšky je vytvořen tak, že zároveň s vyjížděním pístu pro měření výšky je do PLC posílána hodnota v rozsahu 0-27648, přičemž hodnota 27648 odpovídá zasetému pístu. V případě puku nacházejícího se pod pístem, je pomocí komponenty multiplexer odeslána do PLC hodnota přes zpoždění prvního řádu zajišťující plynulou změnu výšky. Výška pro jednotlivé puky byla přepočtena a nastavena v rozsahu 2,1-2,9 cm, kdy hodnotě 27648 odpovídá výška 5 cm. Vysunutím pístu pro měření výšky je také aktivován „stopper“ nacházející se pod ním. Vysunutí jistě závory dojde k aktivaci „stopper“ na nastavených pozicích pro zabránění materiálu v dalším posunu. Pro simulaci vyrážeče je využita komponenta „90DegreeTransfer“, která při jeho vyjetí simuluje sílu pro přesun puku na druhý dopravník. Lineární pohyby pohonů, včetně jejich snímačů koncových poloh, vyjma motorů, jsou

4. Návrh simulačního modelu

simulovány, prostřednictvím komponenty „DriveV1“. Pomocí grafický nástrojů s využitím funkce animace je tento pohyb i vizualizován.



Obrázek 4.7: Simulační model testovacího pracoviště

4.3 Pracoviště procesní



Obrázek 4.8: *Laboratorní model procesního pracoviště*

Procesní pracoviště přesouvá za pomoci chapadel materiál z předchozího pracoviště na jednu z 8 pozic nacházejících se na karuselu. Chapadla jsou poháněna pneumaticky a obsahují 4 snímače informující o stavu pohonu. Karusel je otáčen motorem přes třecí převod. Pod karuselem se nachází píst pro jeho aretaci zajišťující správnou pozici. O vhodné pozici informuje indukční snímač karuselu. Práce na karuselu simuluje operaci vrtání do puku, operaci ražení pneumatickým válcem s pružinovým mechanismem a operaci čištění ofukem. Na pracovišti se také nachází optický snímač puku pod každým chapadlem.

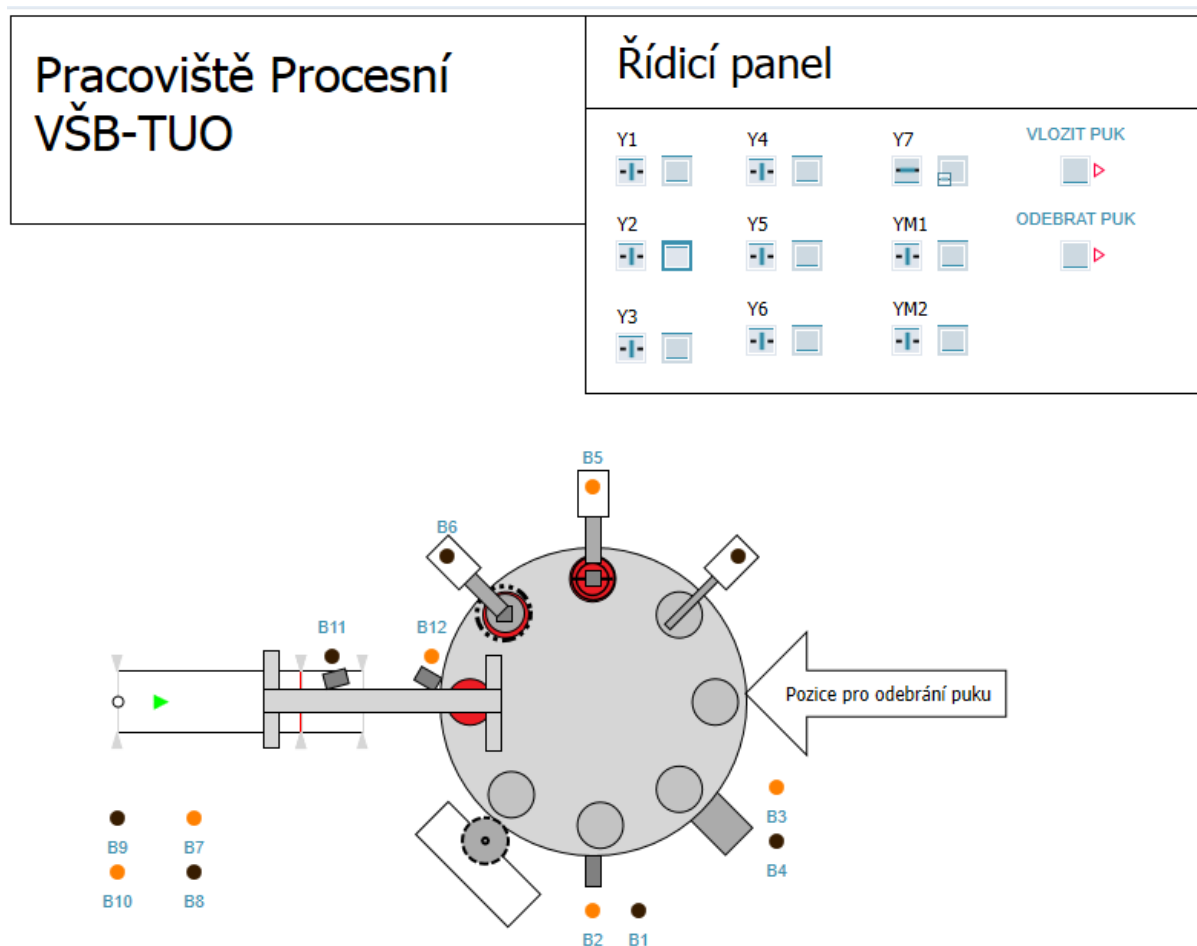
Tabulka 4.3: *Vstupní a výstupní signály procesního pracoviště*

Vstupy			
Symbolický název	Datový typ	Adresa	Komentář
B1	Bool	%I0.0	Rezerva
B2	Bool	%I0.1	Indukční snímač karuselu
B3	Bool	%I0.2	Karusel zaaretován
B4	Bool	%I0.3	Karusel nearetován
B5	Bool	%I0.4	Razník vysunut
B6	Bool	%I0.5	Vrtačka vysunuta
B7	Bool	%I0.6	Chapadla nahoře
B8	Bool	%I0.7	Chapadla dole
B9	Bool	%I1.0	Chapadla uvnitř
B10	Bool	%I1.1	Chapadla vně
B11	Bool	%I1.2	Puk na předchozím stanovišti
B12	Bool	%I1.3	Puk pod chapadly
Výstupy			
Symbolický název	Datový typ	Adresa	Komentář
Y1	Bool	%Q0.0	Aktivace karuselu
Y2	Bool	%Q0.1	Vysunutí razičky
Y3	Bool	%Q0.2	Vysunutí vrtačky
Y4	Bool	%Q0.3	Chapadla dolů
Y5	Bool	%Q0.4	Chapadla otočit
Y6	Bool	%Q0.5	Chapadla sevřena
Y7	Bool	%Q0.6	Aktivace ofuku
YM1	Bool	%Q0.7	Motor karuselu
YM2	Bool	%Q1.0	Motor vrtačky

Simulační model procesního pracoviště obsahuje dvě komponenty dopravníku z knihovny „CONTEC“, které představují část předchozího pracoviště, pro přesunutí puku pod chapadla. Po splnění sekvence příkazu pro uchycení a vyzvednutí puku, je „stopper“ na 0,5 vteřiny deaktivován a prostřednictvím základní komponenty dopravníku, puk z pásu odstraněn. Následně se zobrazí grafický objekt představující puk seskupený s grafickým objektem chapadel. Všechny pohony chapadel jsou simulovány pomocí 3 komponent „DriveV1“. Vizualizace pohybu chapadel je provedena pomocí funkce animace, tak aby bylo možné snadně sledovat každou jejich operaci. Pro umístění puku na karusel musí být splněna posloupnost příkazu a podle aktuální pozice karuselu je s pomocí triviální binární logiky včetně komponenty „RS_FF“ změněna barva slotu na karuselu představující puk a zároveň puk z chapadel zmizí. Pro správnou funkčnost se musí karusel nacházet ve správné pozici. V případě výskytu puku v pozici pro odebrání puku, po stisku tlačítka „ODEBRAT PUK“ proběhne resetování slotu dle pozice karuselu a puk bude odebrán. Motor pohánějící karusel je řízen pomocí komponenty „DriveP1“. Aktuální pozice je získána integrací rychlosti motoru v rozsahu 0-360. Pro

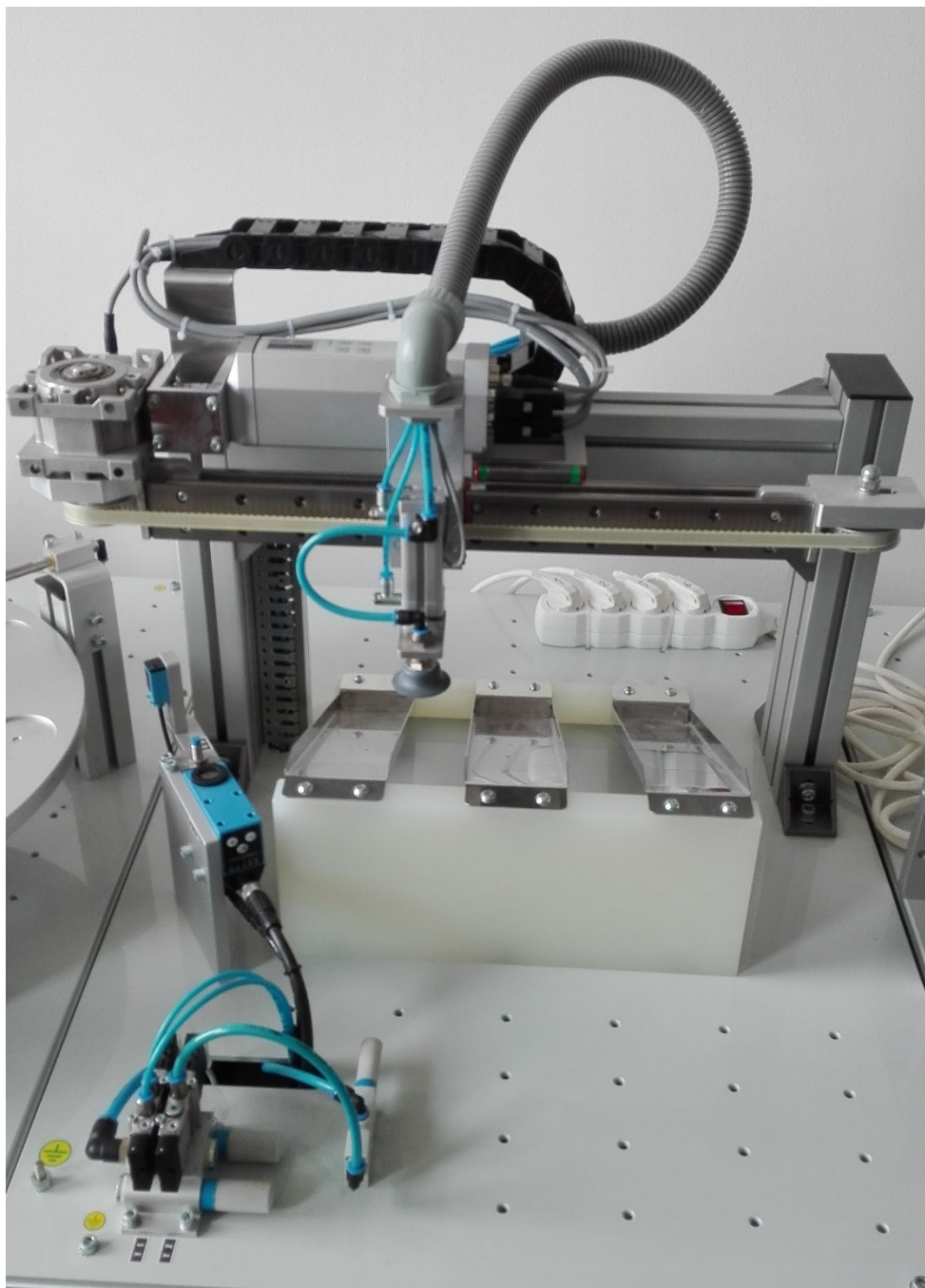
4. Návrh simulačního modelu

aktuální pozici bylo potřeba využít i komponent „selection“ pro upravení polohy v případě aretace karuselu. V tomto modelu je využíván pouze červený puk, jelikož zde není barva puku důležitá. Píst vrtačky, razička a aretace jsou simulovány pomocí komponenty „DriveV1“. Motor vrtačky je simulován přes „DriveP1“. Vrtačka i aretace jsou podrobně vizualizovány ve složce projektu „Detail“.



Obrázek 4.9: Simulační model procesního pracoviště

4.4 Pracoviště manipulační



Obrázek 4.10: *Laboratorní model manipulačního pracoviště*

Manipulační pracoviště realizuje přesun puku z karuselu, který je součástí předchozí úlohy, do jednoho ze skluzů umístěného pod servopohonem nebo na následující pracoviště. Puk je přesunut za pomoci servopohonu s manipulační jednotkou prostřednictvím přísavky. Před pohybem servopohonu je potřeba povolit Enable-bit. Pozice pro přesun servopohonu je nastavena pomocí logického stavu 4 bitových vstupů. Po nastavení jedné z možných pozic je servopohon přemístěn přivedením impulsu na Start-bit. V testovací pozici se nachází snímač barev a snímač magnetických vlastností pro získání informací o přemísťovaném materiálu.

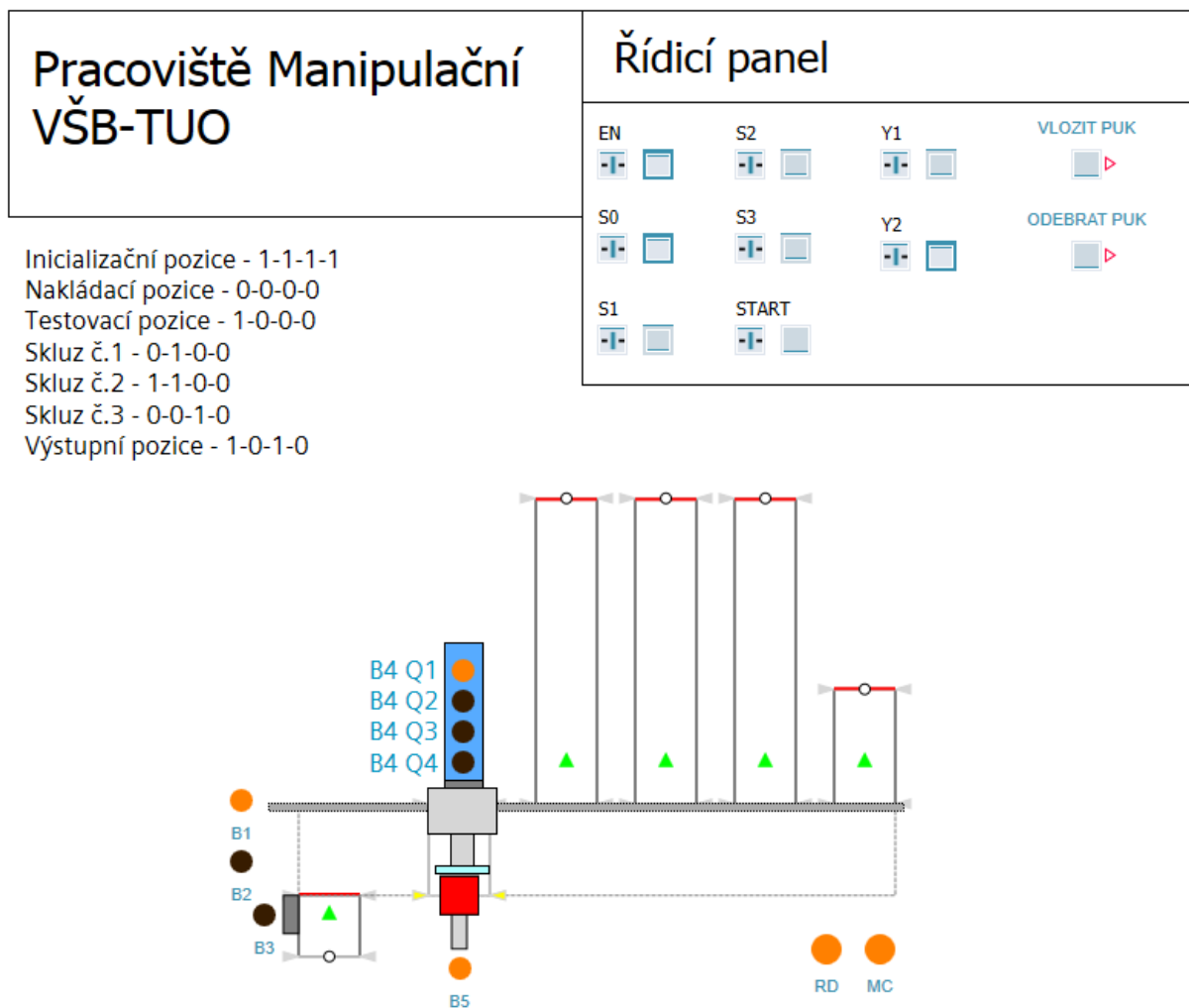
Tabulka 4.4: *Vstupní a výstupní signály manipulačního pracoviště*

Vstupy			
Symbolický název	Datový typ	Adresa	Komentář
RD	Bool	%I0.0	Signalizace připravenosti
MC	Bool	%I0.1	Dokončen pohyb servopohonu
B1	Bool	%I0.2	Píst přísavky zasunut
B2	Bool	%I0.3	Píst přísavky vysunut
B3	Bool	%I0.4	Přítomnost puku na vstupu
B4 Q1	Bool	%I0.5	Snímač barvy - červená
B4 Q2	Bool	%I0.6	Snímač barvy - modrá
B4 Q3	Bool	%I0.7	Snímač barvy - stříbrná
B4 Q4	Bool	%I1.0	Snímač barvy - nespecifikováno
B5	Bool	%I1.1	Snímač magnetických vlastností
Výstupy			
Symbolický název	Datový typ	Adresa	Komentář
S0	Bool	%Q0.0	Servopohon - bit0
S1	Bool	%Q0.1	Servopohon - bit1
S2	Bool	%Q0.2	Servopohon - bit2
S3	Bool	%Q0.3	Servopohon - bit3
START	Bool	%Q0.4	Servopohon - Start-bit
EN	Bool	%Q0.5	Servopohon - Enable-bit
Y1	Bool	%Q0.6	Aktivace pístu s přísavkou
Y2	Bool	%Q0.7	Aktivace přísavky

Celý model je vytvořen pomocí komponent pro dopravníkový pás z knihovny „CONTEC“. U skluzů a výstupní pozice je rychlost nastavená na 100 % po celou dobu simulace, jelikož puk klouže na konečnou pozici příčinou gravitace. Přidržením tlačítka „ODEBRAT PUK“ je „stopper“ na skluzech a výstupní pozici deaktivován. Servopohon je vytvořen komponentou „TransferCarriage“, která umožňuje horizontálně přesouvat materiál. Vstupní nebo výstupní pozice můžeme měnit přidržením klávesy ALT a stiskem levého tlačítka myši na propojovací bod společně s následným přetažením na žádanou pozici. Rychlost motoru servopohonu je simulována komponentou „DriveP2“. Následnou integrací je získána uražená dráha servopohonu s maximální nabývací hodnota 1300, což je dostačující. Ve složce pro matematické operace a její podsložce „Servopohon“ je kompletně vyřešená binární logika pro napodobení řízení skutečného servopohonu. Zdvih puku musí proběhnout sekvenčně správně, aby došlo k deaktivaci zadržování puku na 0,4 vteřin v místě nakládací pozice a puk se přesunul na dopravníkovou komponentu simulující servopohon. Přesunem puku na testovací pozici je detekována barva a ověřena magnetická vlastnost, stejným způsobem jako u předchozích pracovišť. Položení puku musí být provedeno také sekvenčně správně, čímž dojde k nastavení 100 % rychlosti po dobu 3 vteřin u dopravníkové komponenty simulující servopohon a umožní puku zajet do jednoho ze skluzů nebo

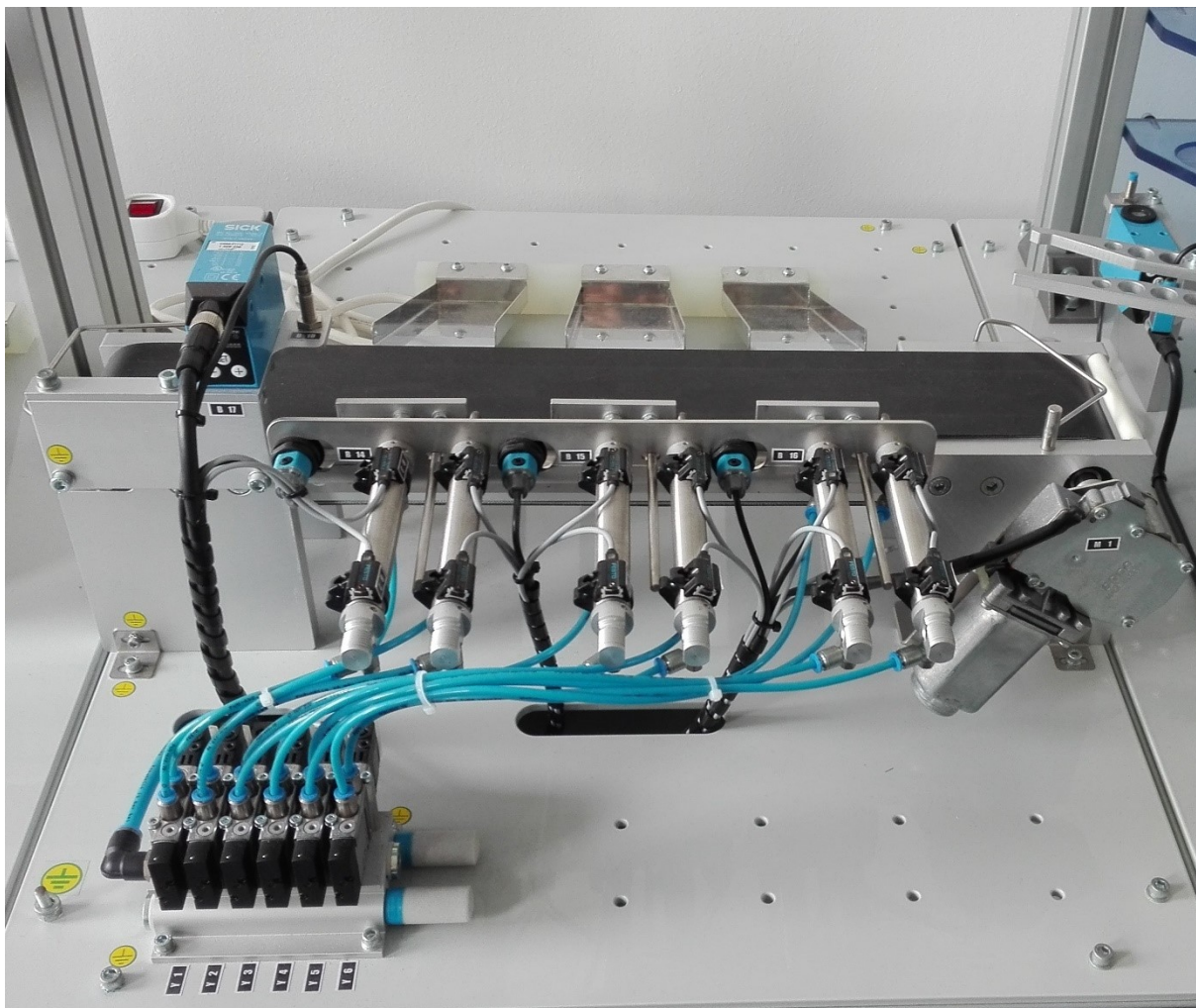
4. Návrh simulačního modelu

na výstupní pozici. Kontrol „RD“ svítí v případě, že je rychlost servopohonu nulová. Kontrol „MC“ se rozsvítí vždy po dobu 2 vteřin po dokončení pohybu servopohonu.



Obrázek 4.11: Simulační model manipulačního pracoviště

4.5 Pracoviště třídící



Obrázek 4.12: *Laboratorní model třídícího pracoviště*

Účelem třídící pracoviště je roztrdit puky do jednotlivých skluzů, dle zjištěných vlastností na začátku dopravníku. Model se skládá z testovací oblasti, ve které se nachází snímač barvy a snímač magnetických vlastností a z 3 třídících oblastí, obsahující optický snímač a pneumaticky poháněný vyrážecí a závora včetně jejich koncových snímačů. Materiál je přesouván pomocí pásového dopravníku poháněného elektromotorem. Úloha obsahuje i indukční snímač rotace pásu, stejně jako u distribučního a testovacího pracoviště.

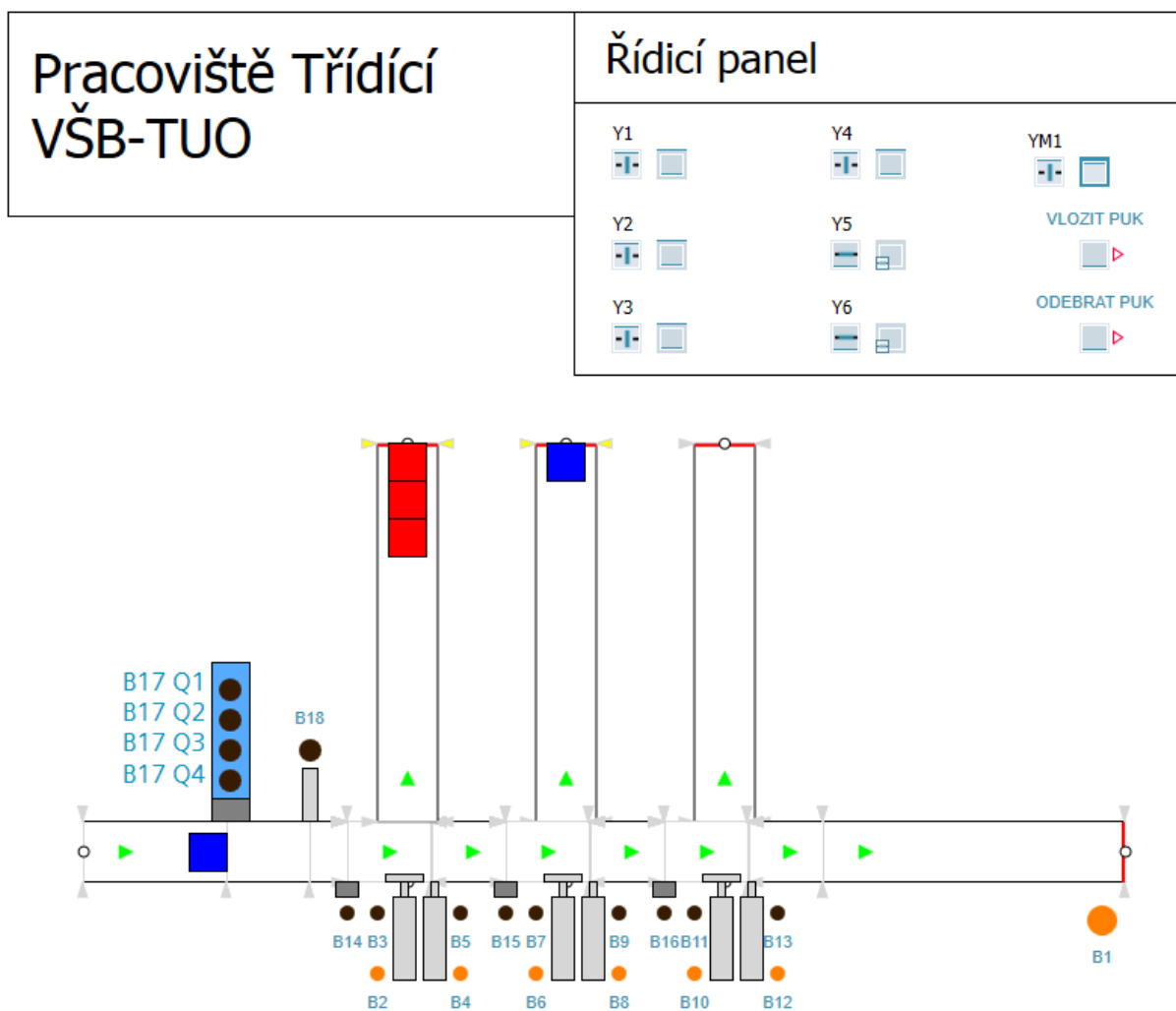
Tabulka 4.5: *Vstupní a výstupní signály třídícího pracoviště*

Vstupy			
Symbolický název	Datový typ	Adresa	Komentář
B1	Bool	%I0.0	Indukční snímač rotace pásu
B2	Bool	%I0.1	Vyrážecí 1 zasunut
B3	Bool	%I0.2	Vyrážecí 1 vysunut
B4	Bool	%I0.3	Závora 1 zasunuta

4. Návrh simulačního modelu

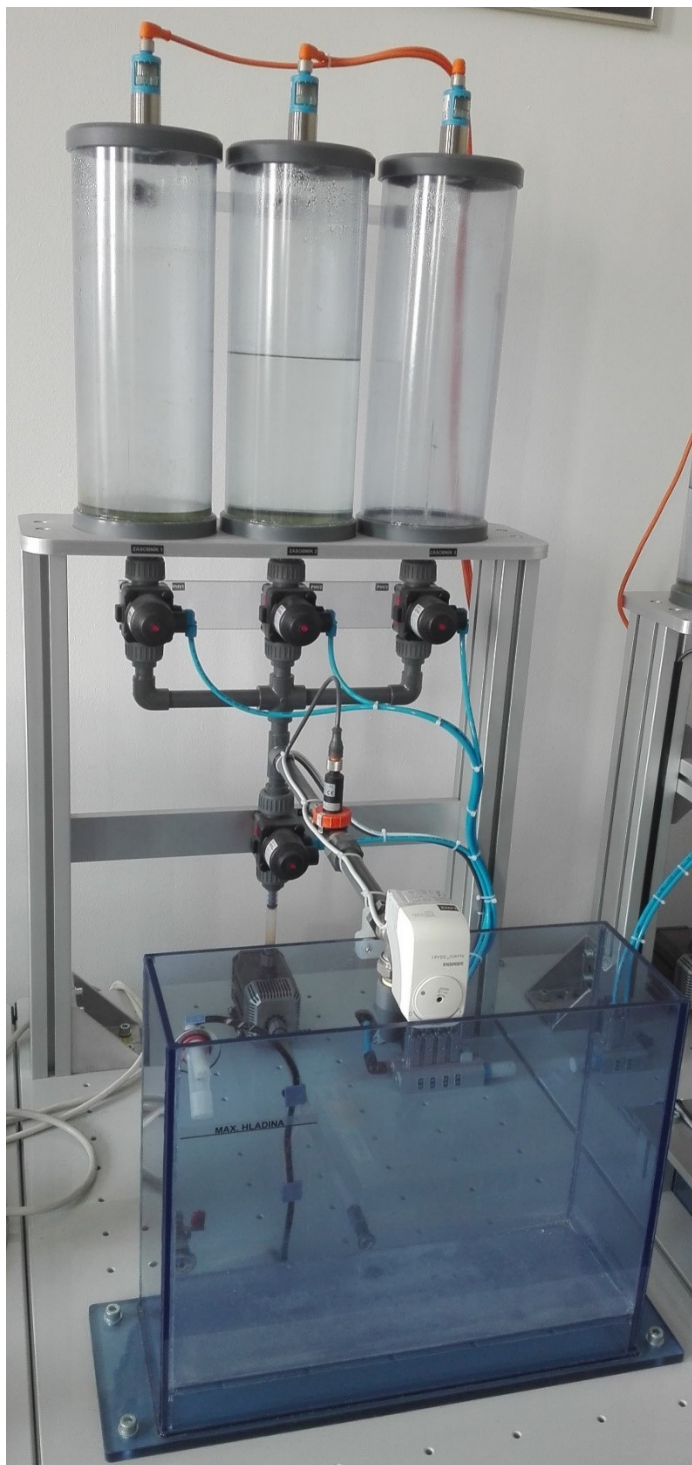
B5	Bool	%I0.4	Závora 1 vysunuta
B6	Bool	%I0.5	Vyrážeč 2 zasunut
B7	Bool	%I0.6	Vyrážeč 2 vysunut
B8	Bool	%I0.7	Závora 2 zasunuta
B9	Bool	%I1.0	Závora 2 vysunuta
B10	Bool	%I1.1	Vyrážeč 3 zasunut
B11	Bool	%I1.2	Vyrážeč 3 vysunut
B12	Bool	%I1.3	Závora 3 zasunuta
B13	Bool	%I1.4	Závora 3 vysunuta
B14	Bool	%I1.5	Snímač přítomnosti 1
B15	Bool	%I1.6	Snímač přítomnosti 2
B16	Bool	%I1.7	Snímač přítomnosti 3
B17 Q1	Bool	%I2.0	Snímač barev - červená
B17 Q2	Bool	%I2.1	Snímač barev - modrá
B17 Q3	Bool	%I2.2	Snímač barev - stříbrná
B17 Q4	Bool	%I2.3	Snímač barev - nespecifikována
B18	Bool	%I2.4	Snímač magnetických vlastností
Výstupy			
Symbolický název	Datový typ	Adresa	Komentář
Y1	Bool	%Q0.0	Aktivace vyrážeče 1
Y2	Bool	%Q0.1	Aktivace závory 1
Y3	Bool	%Q0.2	Aktivace vyrážeče 2
Y4	Bool	%Q0.3	Aktivace závory 2
Y5	Bool	%Q0.4	Aktivace vyrážeče 3
Y6	Bool	%Q0.5	Aktivace závory 3
YM1	Bool	%Q0.6	Aktivace motoru dopravníkového pásu

Simulační model je poskládán z pásových dopravníků nacházejících se v knihovně „CONTEC“. Na začátku komponenty pásu je v parametrech potřeba nastavit 2 senzory, včetně jejich pozic, z kterých budou získávány informace o přemísťovaném objektu. V dodatečných parametrech se musí vepsat typ materiálu a propojit vstupní signál s tlačítkem pro vložení puku na pás. Snímač barev zde funguje stejně jako u předchozích pracovišť, podle ID puku je prostřednictvím komponenty interval poslán logický signál do PLC. Magnetický snímač reaguje na každé liché ID puku. U každého vyrážeče je dopravníková komponenta „90DegreeTransfer“ umožňující přesun puku do skluzu při vysouvání vyrážeče, což reprezentuje sílu vyvinutou při vyrážení. Tato komponenta obsahuje i výstup, kterým zasílá data pro optický snímač přítomnosti. Následující dopravníková komponenta je „Conveyor-S4-Stopper“, kde je aktivován vstup stop při vyjíždění závory. Série pásových komponent je poháněná výstupem z komponenty „DriveV1“. U skluzů je definovaná rychlost 100 %, což představuje gravitační sílu. Pohyb vyrážečů i závor je simulován pomocí komponenty „DriveV1“, která obsahuje i výstupy pro koncové snímače. Integrací rychlosti motoru pásu získáme polohu pootočení motoru pro simulaci indukčního snímače rotace.



Obrázek 4.13: *Simulační model třídícího pracoviště*

4.6 Pracoviště mixovací



Obrázek 4.14: *Laboratorní model mixovacího pracoviště*

Laboratorní model se skládá z 3 zásobníku s kapalinou, které obsahují ultrazvukové snímače pro měření výšky hladiny, z mixovací nádrže včetně plovákového čidla, odstředivého čerpadla, impulsního průtokoměru, analogového elektricky řízeného ventilu a 4 pneumaticky řízených membránových ventilů. Ultrazvukový snímač indikuje výšku aktuální hladiny v rozsahu 0-10 V,

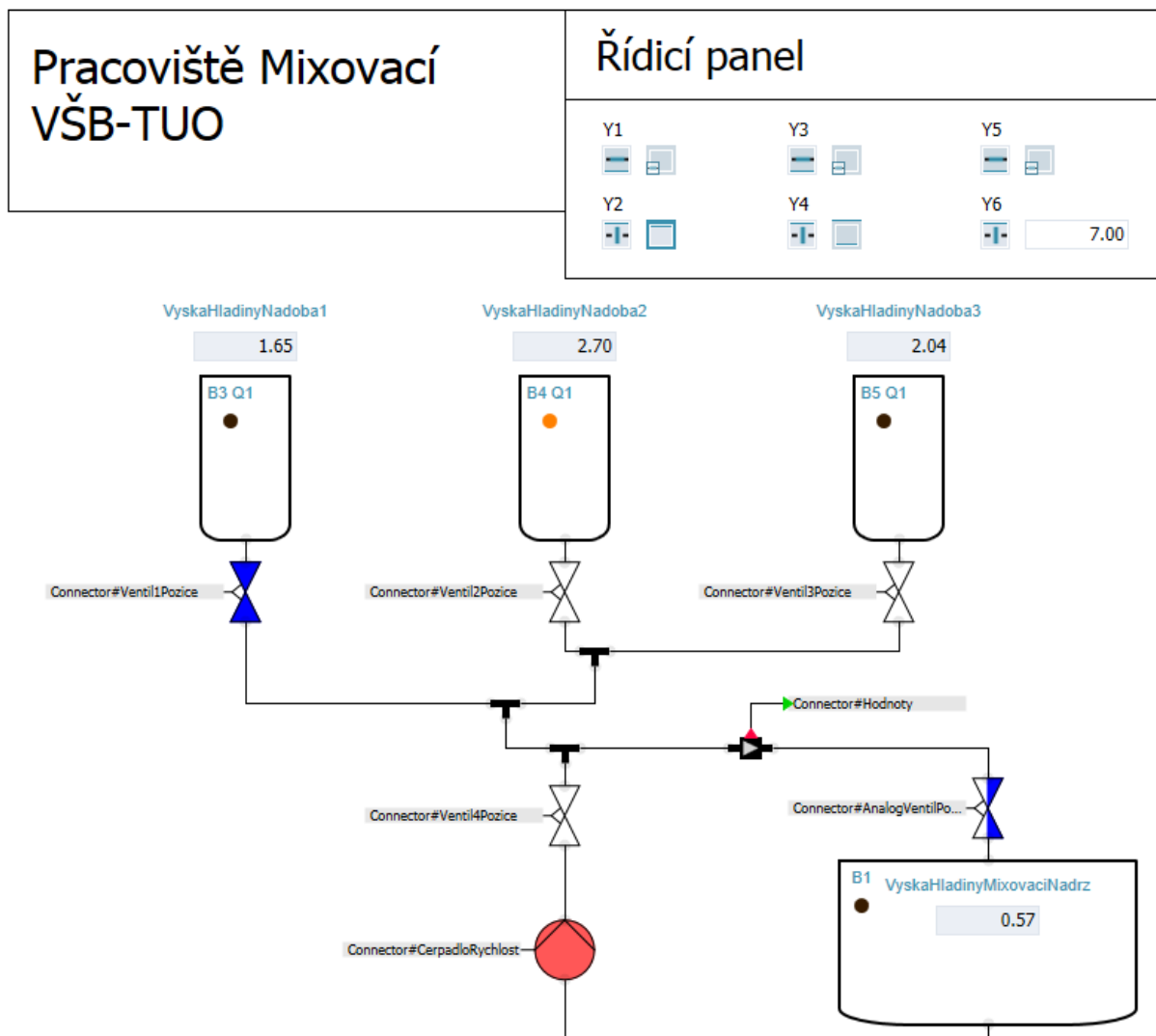
4. Návrh simulačního modelu

přičemž 10 V odpovídá minimální hladině. Analogový ventil je ovládán napětím v rozsahu 0-10 V, kde při hodnotě 10 V je ventil plně otevřen. Průtokoměr má frekvenční výstup 0-505 Hz odpovídajících průtoku 0-6 l/min.

Tabulka 4.6: *Vstupní a výstupní signály mixovacího pracoviště*

Vstupy			
Symbolický název	Datový typ	Adresa	Komentář
B1	Bool	%I0.0	Maximální hladina mixovací nádrže
B2	Bool	%I0.1	Frekvenční výstup průtokoměru
B3 Q1	Bool	%I0.2	Maximální hladina - zásobník č.1
B4 Q1	Bool	%I0.3	Maximální hladina - zásobník č.2
B5 Q1	Bool	%I0.4	Maximální hladina - zásobník č.3
B3 QA	Int	%IW2	Výška hladiny - zásobník č.1
B4 QA	Int	%IW4	Výška hladiny zásobník č.2
B5 QA	Int	%IW6	Výška hladiny zásobník č.3
Výstupy			
Symbolický název	Datový typ	Adresa	Komentář
Y1	Bool	%Q0.0	Otevření ventilu - čerpadlo
Y2	Bool	%Q0.1	Otevření ventilu - zásobník č.1
Y3	Bool	%Q0.2	Otevření ventilu - zásobník č.2
Y4	Bool	%Q0.3	Otevření ventilu - zásobník č.3
Y5	Bool	%Q0.4	Aktivace čerpadla
Y6	Int	%QW2	Analogové otevírání ventilu před mixovací nádrží

Model pro simulaci mixovacího pracoviště byl vytvořen s využitím komponent z knihovny „FLOWNET“. Kapalina u laboratorního modelu je voda, proto byli zvoleny komponenty s již nastavenými fyzikálními parametry pro tuto látku v sekci „WATER.STEAM“. Individuální zásobníky a mixovací nádrž byly vytvořeny komponentou „StorageTankWS“. Model je vytvořen v měřítku 10:1 vůči laboratornímu modelu, simulace tak probíhá více detailněji. Objem zásobníků byl vypočten a nastaven na 5,3 m³ s výškou 3 m., objem nádrže na 16 m³ s výškou 3 m. Tlak v zásobnících je nastaven na 3 bar pro rychlejší průtok vody do nádrže, v které je tlak 1 bar. Na digitální displej je přiveden výstupní signál pro zobrazení výšky hladiny z jednotlivých nádob na vodu. Snímače maximální hladiny se nachází v log. 1 při dosažení výšky 2,55 m. Analogové ultrazvukové snímače cyklicky posílají vstupní signál do PLC v rozsahu 0-27648, kdy při nulové hladině je posílána hodnota 27648. Komponentu ventilu najdeme v již zmíněné knihovně mezi základními prvky. Jejich lineární otevření je řízeno přes pohon „DriveV1“ a v případě analogového ventilu přes pohon „DriveV4“, umožňující nastavit procentuální hodnotu otevření. Čerpadlo představuje komponentu „Pump“, jejíž spouštění probíhá lineárně pomocí komponenty „DriveP1“ Jmenovitý tlak čerpadla je nastaven na 16 bar, pro rychlé načerpání vody zpět do zásobníků. Pro měření průtoku a následnou simulaci průtokoměru bylo využito komponenty „PipeMeasure“ a dle jednoduchých matematických operací a makra „Square“ vygenerován pulzní signál v rozsahu 0-505 Hz. Spojení soustavy je provedeno pomocí T-propojek.



Obrázek 4.15: Simulační model mixovacího pracoviště

5. Realizace řídicí aplikace

Řídicí aplikace pro jednotlivé modely byly vytvořeny ve vývojovém prostředí TIA Portal V13 v programovacím jazyce kontaktních schémat. Programy byly vytvořeny z návrhu Petriho sítí s pomocnými stavovými paměťovými bity zajišťující sekvenční probíhání kódu. U každého projektu je vytvořena hardware konfigurace odpovídající skutečnému PLC nacházejícího se v laboratoři programovatelných automatů. Pro aplikace byl využit řídicí systém Simatic S7-1500 značky Siemens, který je pro naše aplikace zcela postačující.



Obrázek 5.1: Hardware konfigurace PLC

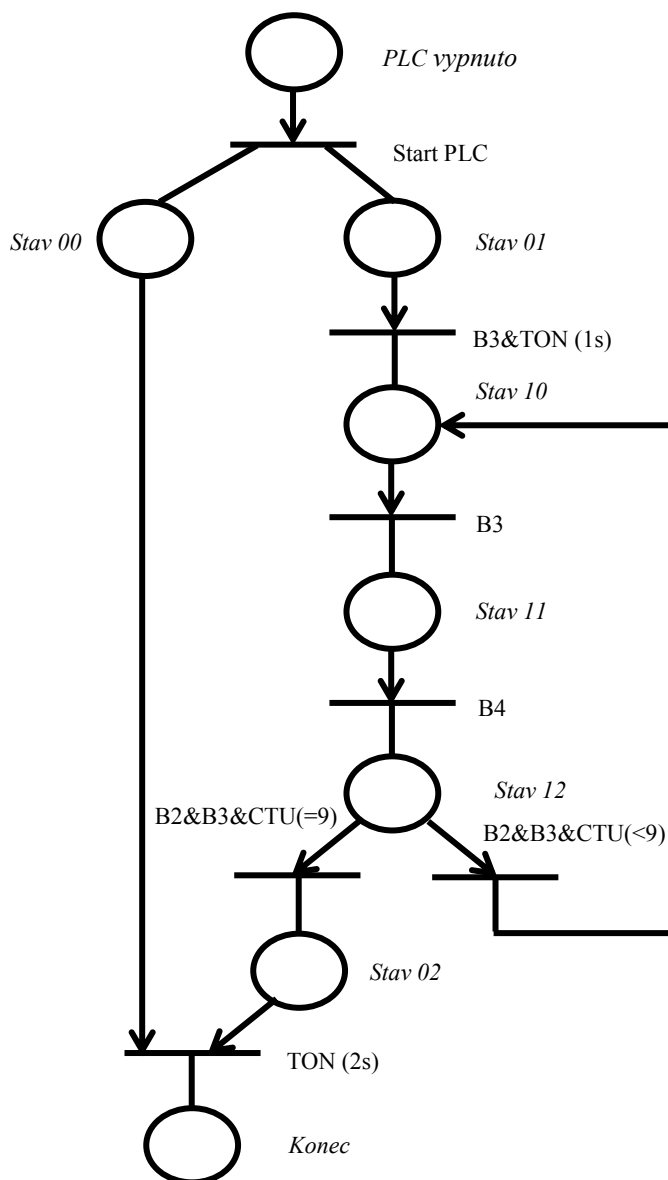
Po úspěšném vložení všech odpovídajících typů modulů, bylo potřeba nastavit, ve vlastnostech CPU, správnou IP adresu odpovídající danému PLC, zajišťující správnou komunikaci. V sekci protekce bylo nutné zaškrtnout možnost volby pro úplný přístup k PLC a svolit přístup pro „PUT/GET“ komunikaci od vzdáleného zařízení, umožňující vzdálené čtení i zápis dat.

Tabulka 5.1: Hardware konfigurace PLC a adresy

Slot	Modul	Číslo výrobku	Firmware	Vstupní adresy	Výstupní adresy
0	PS 25W 24VDC	6ES7 505-0KA00-0AB0	V1.0		
1	CPU 1511-1 PN	6ES7 511-1AK00-0AB0	V1.1		
2	CP 1543-1	6GK7 543-1AX00-0XE0	V1.1		
3	DI 16x24VDC HF	6ES7 521-1BH00-0AB0	V2.0	0..1	
4	DQ 16x24VDC/0.5A ST	6ES7 522-1BH00-0AB0	V2.0		0..1
5	AI 8xU/I/RTD/TC ST	6ES7 531-7KF00-0AB0	V2.0	2..17	
6	AQ 4xU/I ST	6ES7 532-5HD00-0AB0	V2.0		2..9

5.1 Pracoviště distribuční

Aplikace je vytvořena tak, aby podavač vyřázel postupně všech 9 puků nacházejících se v zásobníku a ty byly přemísťovány pomocí dopravníku poháněným elektrickým motorem na konec pracoviště. Po devíti impulsích zaslanych optickým snímačem je motor pásu vypnut se zpožděním, což umožní poslednímu puku dorazit na konec pásu.

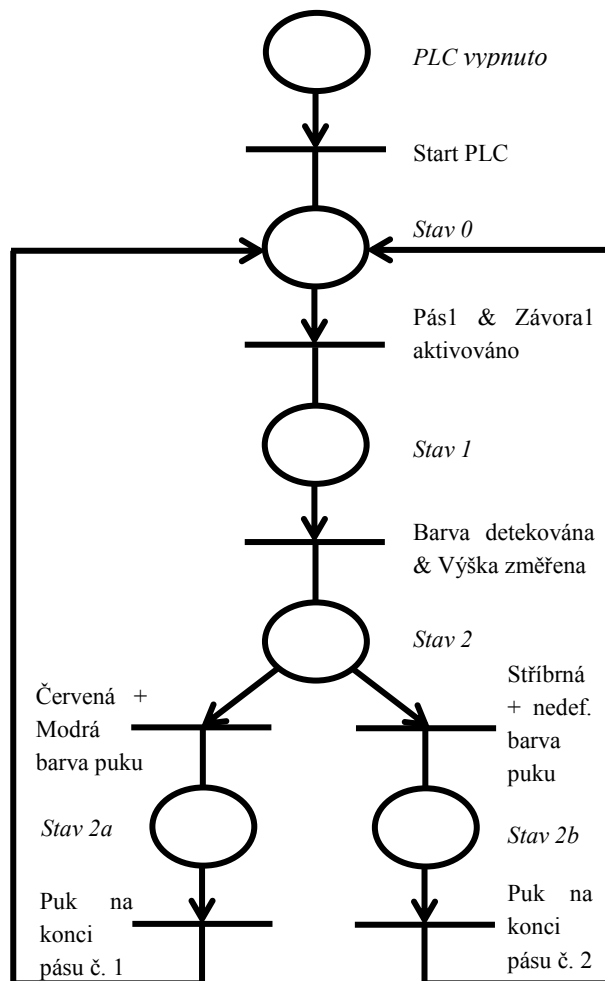


Obrázek 5.2: Návrh Petriho sítě pro distribuční pracoviště

Řídicí aplikace pro distribuční pracoviště byla vytvořena pomocí návrhu Petriho sítě. V OB100, což představuje část programu, která proběhne pouze při startu PLC, je nastaven stav, spouštějící motor dopravníkového pásu do log. 1, a stav aktivující cyklické probíhání kódu programu s jednovteřinovým zpožděním. Hlavní program pro vyřazení puků probíhá ve funkčním bloku, tak dlouho dokud není vyraženo 9 puků, poté je s dvouvteřinovým zpožděním přes časovač TON resetován stav aktivující motor dopravníku.

5.2 Pracoviště testovací

Pracoviště je naprogramováno, tak aby po detekci jednoho ze čtyř vstupů snímače barev zahájila sekvenci vykonávání kódu. Nejdříve proběhne vysunutí první závory a začne měření výšky vyjetím pístu, který po 2 vteřinách zajede zpět. Závora zajede a po detekci puku optickým snímačem proběhne výběr jedné ze dvou následujících sekcí. Při detekci červené nebo modré barvy je puk následně přemístěn na konec pásu. Při detekci stříbrné nebo nedefinované barvy je puk prostřednictvím druhé závory a vyrážече přemístěn na druhý dopravník, kde je dopraven až na konec pásu. Následně je očekáván další puk snímačem barvy pro další proběhnutí programu.

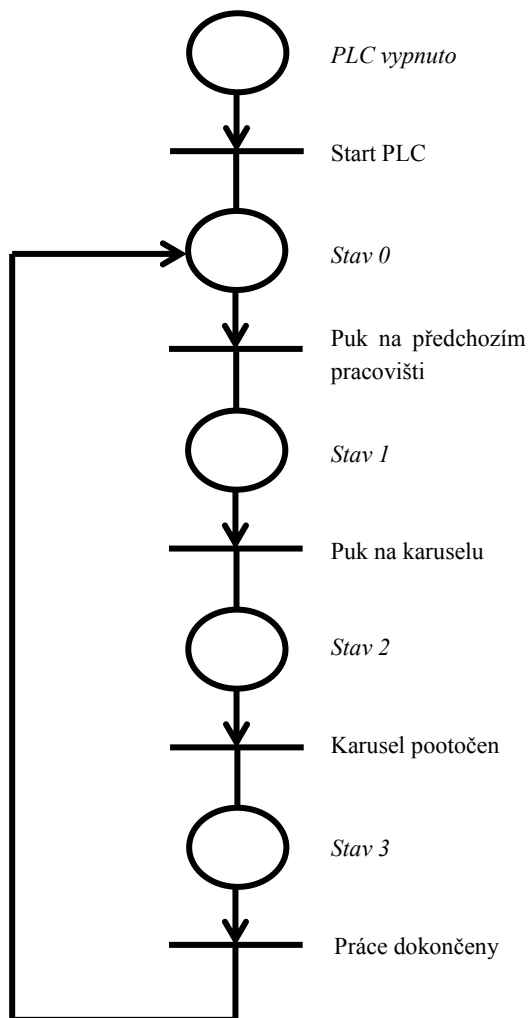


Obrázek 5.3: Návrh Petriho sítě pro testovací pracoviště

Pomocí návrhu Petriho sítě byl realizován program, který po startu PLC aktivuje funkční blok, který uvede do pohybu hlavní dopravník a vysune první závoru. Následně je aktivován další funkční blok pro získání fyzikálních vlastností puku. V případě detekce červené nebo modré barvy je puk přemístěn na konec hlavního dopravníku pomocí třetího funkčního bloku. V případě stříbrné nebo nedefinované barvy je puk přemístěn na druhý dopravník díky operacím ve čtvrtém funkčním bloku. Jednoduchou úpravou programu lze docílit třízení puku dle jiných parametrů. Po proběhnutí kódu ve třetím nebo čtvrtém funkčním bloku je vyresetována paměť s vlastnostmi puku a aktivován druhý funkční blok pro opakování cyklu.

5.3 Pracoviště procesní

Řídicí program byl navržen a zrealizován tak, aby po detekci puku na předchozím pracovišti optickým snímačem byla spuštěna posloupnost příkazů zajišťující přemístění puku na karusel za pomoci pneumaticky poháněných chapadel. Po správném přesunu je provedeno pootočení karuselu o jednu pozici, v případě nesprávné pozice proběhne inicializace. Na konec jsou sekvenčně provedeny práce s vrtačkou, razničkou a ofukem, po nichž může dojít k přemístění dalšího puku na karusel.

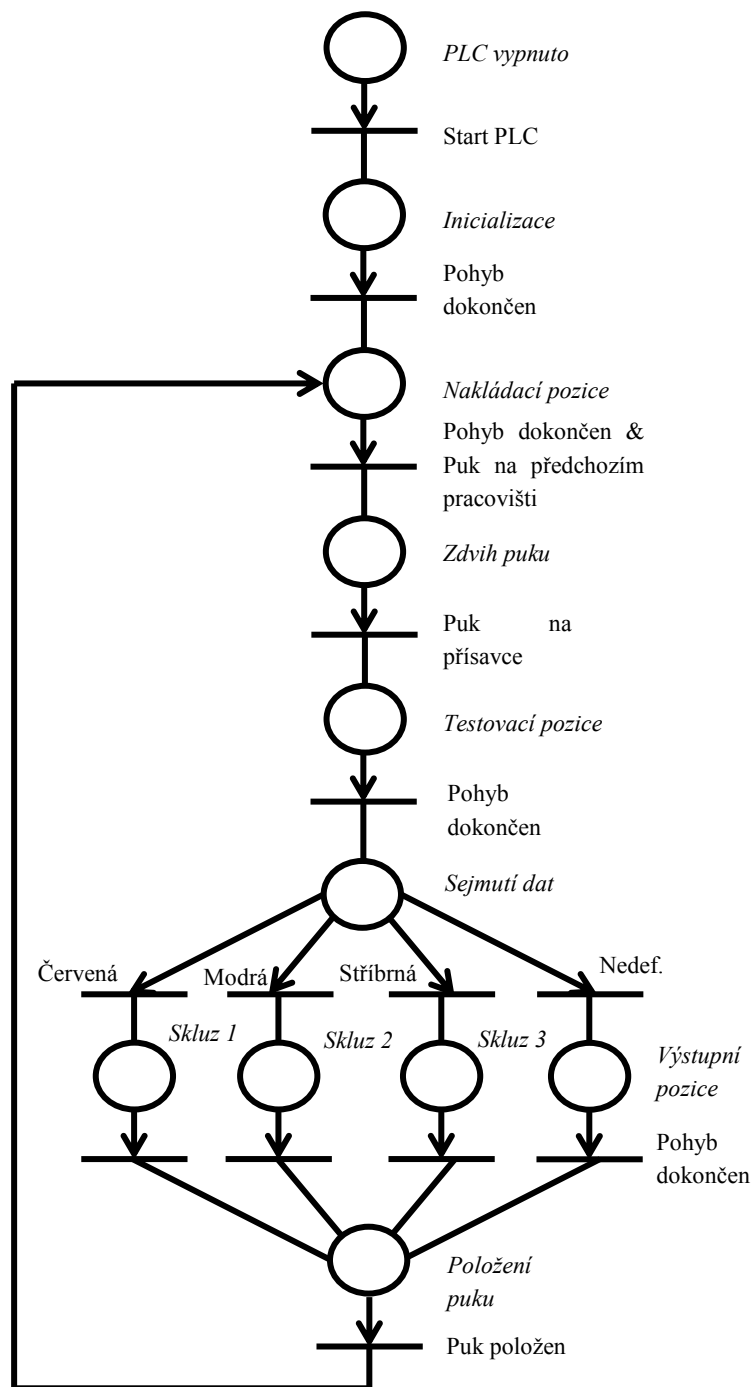


Obrázek 5.4: Návrh Petriho sítě pro procesní pracoviště

Z návrhu Petriho sítě pro procesní pracoviště byl vytvořen program skládající se z funkčních bloků pro větší přehlednost kódu. Po startu PLC je aktivován první stav, který, v případě nacházejícího se puku na konci pásu předchozího pracoviště, aktivuje postupně následující stavy v prvním funkčním bloku. Při detekci puku optickým snímačem jsou spuštěny chapadla do spodní polohy a za jejich pomoci je sevřen puk. Chapadla jsou poté otočena o 180° a po jejich spuštění a rozevření je puk umístěn na karusel. Chapadla jsou vrácena zpět do původní polohy. V druhém funkčním bloku se nachází program pro otočení karuselu o jednu pozici. Karusel je prvně aktivován a poté je zapnut motor pohánějící karusel. Při detekci indukčního snímače, pro zajištění správné pozice, je motor vypnut a karusel zaaretován. V posledním funkčním bloku jsou řízeny práce na puku. Spustí se motor vrtačky a po 1 vteřině vrtačka vyjede a ihned zajede, po 1 vteřině je vypnut i motor. Následující prací je aktivace razičky, po jejímž proběhnutí je spuštěn na 2 vteřiny ofuk.

5.4 Pracoviště manipulační

Pracoviště je ovládáno pomocí PLC, ve kterém se nachází program, zajišťující přesun puku pomocí servopohonu s manipulační jednotkou. Puk je vyzvednut z karuselu, které je součástí předchozího pracoviště, za pomoci přísavky. Materiál je přemístěn nejprve na testovací pozici obsahující snímač barvy, podle kterého je zvolena jedna z výstupních pozic, do které je puk přemístěn a položen.



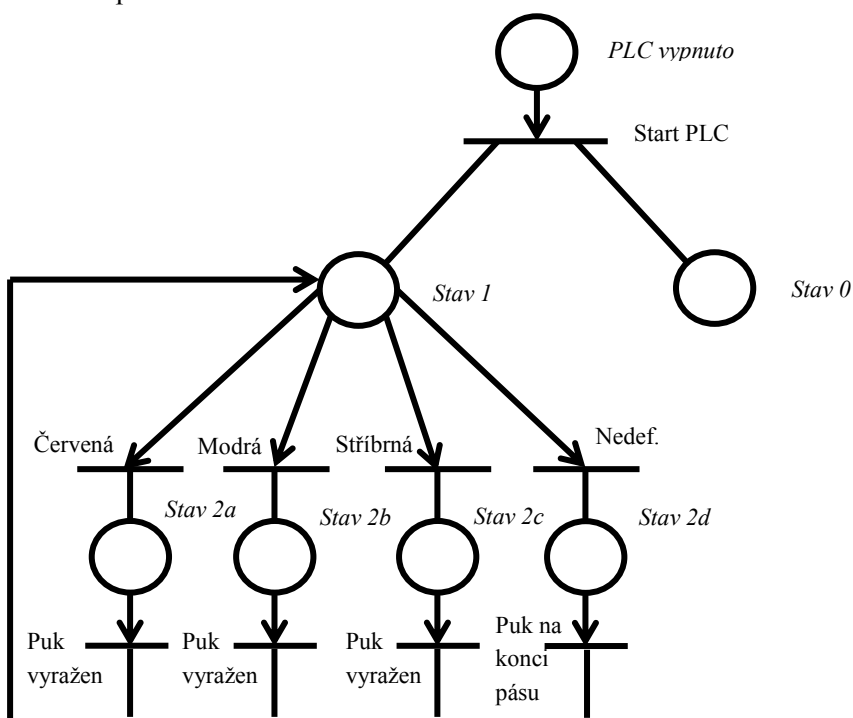
Obrázek 5.5: Návrh Petriho sítě pro manipulační pracoviště

Program je složen z funkčních bloků pro přesun puku, zdvih a položení puku. Po startu PLC je aktivován výstup Enable-bit a počáteční stav zajišťující přesun do inicializační pozice. Po dokončení

pohybu je servopohon přemístěn na nakládací pozici, kde proběhne zdvižení puku vyjetím pístu s přísavkou a aktivace přísavky pro přísátí puku, aby nedošlo k poškození, zajede píst zpět. Přesunem na testovací pozici je rozeznána barva puku a proběhne přesun na jednu z výstupních pozic, odkud je puk položen vysunutím pístu a deaktivací přísavky. Servopohon zajede zpět do nakládací pozice a očekává signál z optického snímače indikující puk na předchozím pracovišti.

5.5 Pracoviště třídící

Pracoviště na svém začátku obsahuje snímače pro detekci barvy a magnetických vlastností puku. Aplikace je navržena tak, aby rozdělila puky do jednotlivých skluzů, dle nasnímané barvy, pomocí závor a vyrážeců. V případě nedefinované barvy je puk přemístěn na konec dopravníku. Puk je po pracovišti přesouván za pomoci dopravníku.

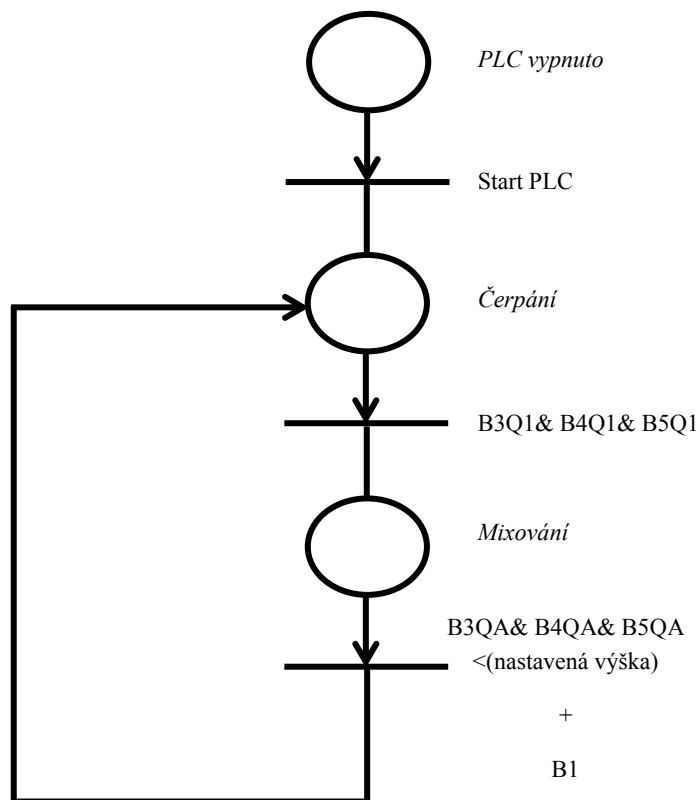


Obrázek 5.6: Návrh Petriho sítě pro třídící pracoviště

Po startu PLC jsou aktivovány dva stavy, jeden pro spuštění motoru pohánějící pás a druhý aktivující funkční blok s programem pro detekci fyzikálních vlastností puku. Podle získaných informací je provedena aktivace následujícího funkčního bloku, ve kterém je po detekci optickým snímačem přítomnosti vysunuta závara a po jedné vteřině i vyrážec, zajišťující kolmý přesun puku. Po vyražení puku je závara i vyrážec zasunut zpět do výchozí polohy. V případě detekce nedefinované barvy je puk přesunut na konec pracoviště a po 10 vteřinách může proběhnout detekce nového puku.

5.6 Pracoviště mixovací

Úkolem navrhovaného programu je cyklicky řídit čerpání vody z mixovací nádrže pomocí čerpadla a následné upouštění kapalin z jednotlivých zásobníků do zadané výšky hladiny pro každý zásobník.



Obrázek 5.7: Návrh Petriho sítě pro mixovací pracoviště

Po startu PLC je spuštěno čerpání vody, v případě že výška hladiny alespoň jednoho zásobníku je pod maximální hodnotou. Čerpání probíhá nejprve otevřením příslušného ventilu pro zásobník, kde ještě nebyla dosažena maximální hladina a následným uzavřením ventilu pro čerpadlo a analogový ventil. Následně je spuštěno čerpadlo a po jedné vteřině je otevřen ventil nacházející se nad ním. Po dosažení maximální hladiny pro jednotlivý zásobník je jejich přívodní ventil uzavřen. S naplněním všech zásobníků je ventil čerpadla uzavřen a čerpadlo po 1 vteřině vypnuto. Mixování kapalin probíhá tak, že je otevřen ventil pro zásobník, který obsahuje výšku hladiny nad nastavenou hodnotu a v případě nenaplněné mixovací nádrže je otevřen analogový ventil, který je po vypuštění všech zásobníků do požadované výšky uzavřen.

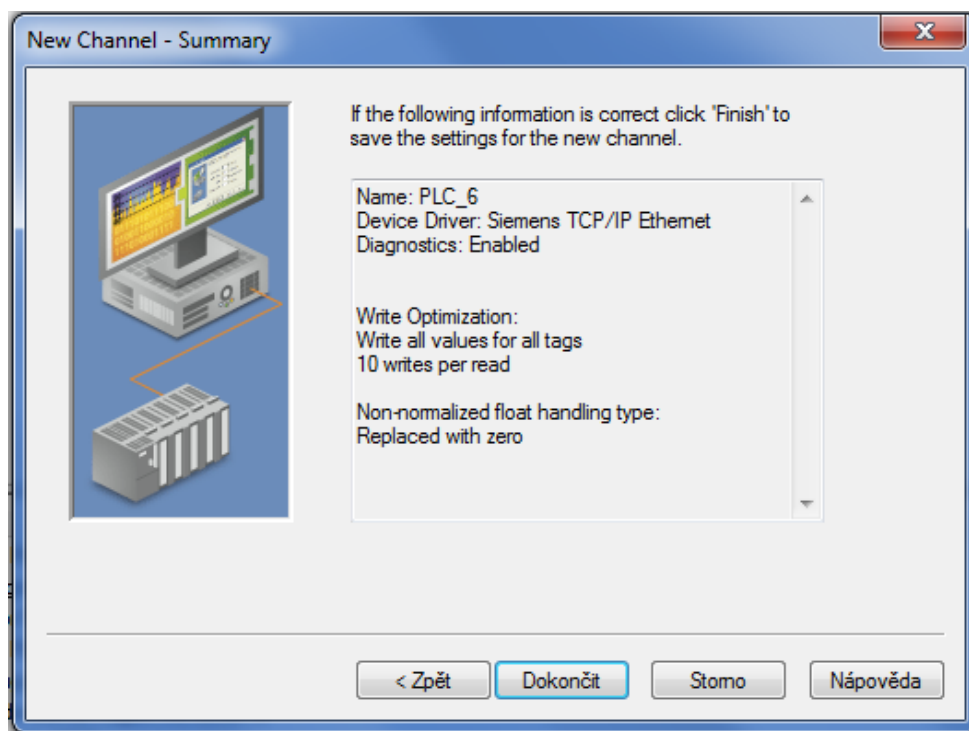
6. Ověření funkčnosti

Pro otestování simulačního modelu a řídicí aplikace je možné zvolit několik způsobů komunikace, které SIMIT nabízí. PROFIBUS DP a PROFINET IO je nejlepší volbou komunikace, během které automat pracuje jako se skutečným systémem, jedná se o HiL simulaci. Bohužel lze tento způsob využít pouze s dostupným interface modulem SIMBA PROFIBUS nebo SIMBA PNIO. PRODAVE umožňuje komunikovat pouze s CPU a neumožňuje pracovat se vstupními a výstupními moduly, jedná se o PiL simulaci. Pro jeho použití je potřeba mít nainstalován software SIMATIC PRODAVE ve verzi 6.2. Simulace s využitím PLCSIM je nazývána jako SiL simulace. Jedná se o vhodný způsob pro simulaci s minimální potřebou hardware, pro simulaci vystačí pouze PC. Nevýhodou je nekompatibilita SIMIT s novějšími verzemi PLCSIM, pracuje pouze s verzí PLCSIM 5.4 SP5, která je spustitelná pouze na starších OS jako je Windows XP nebo Windows 7. [32]

Jedním z dalších možných způsobů komunikace, zároveň vybraným v této práci, je komunikace za pomoci průmyslového standartu OPC DA ve verzi 3.0. SIMIT je nastaven jako OPC klient, získávající data z OPC serveru, který je vytvořen na stejném PC, na kterém běží SIMIT. Tato komunikace využívá techniku HiL.

6.1 Nastavení OPC serveru

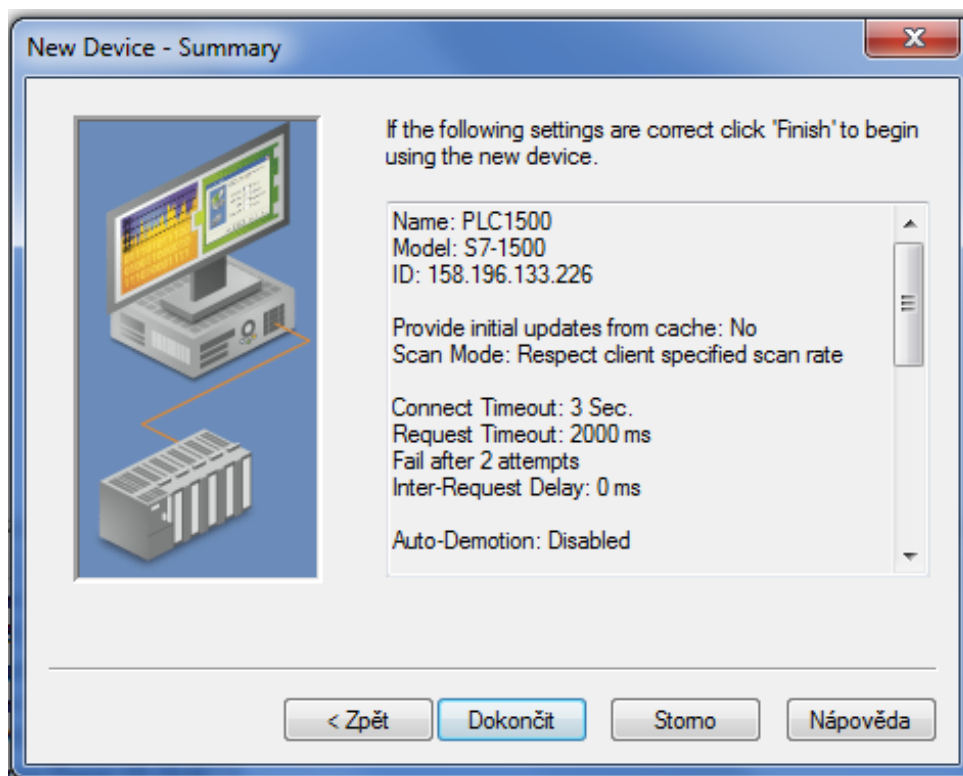
OPC server je vytvořen v programu KEPServerEX, umožňující komunikaci po dobu dvou hodin zdarma. Pro správnou konfiguraci OPC serveru bylo potřeba spustit „KEPServerEX 5 Configuration“ a vytvořit nový kanál, v tomto případě byl pojmenován PLC_6 a jako ovladač zařízení zvolen Siemens TCP/IP Ethernet. Ostatní možnosti už nebylo potřeba měnit a stačilo je ponechat defaultní.



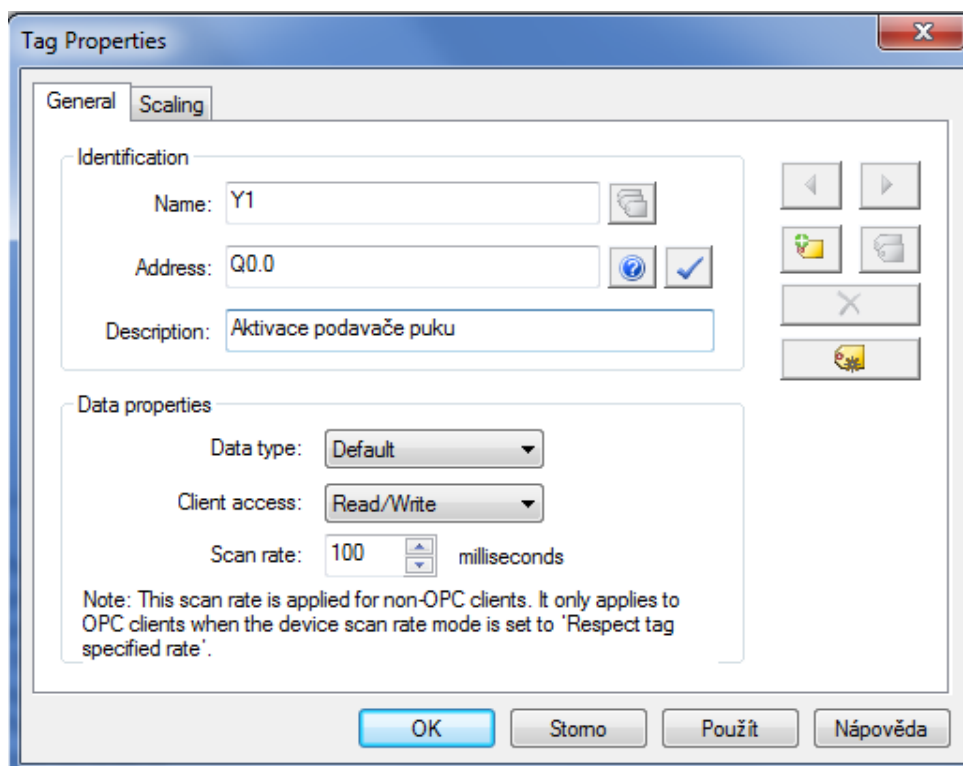
Obrázek 6.1: Přidání nového kanálu v KEPServerEX

6. Ověření funkčnosti

Dalším nezbytným krokem bylo přidat zařízení do již vytvořeného kanálu. Zařízení bylo pojmenováno PLC1500 a jako model vybrán S7-1500, což představuje používané PLC. Do oblasti ID zařízení je nastavena IP adresa automatu. Další možnosti nebylo nezbytné měnit a jejich nastavení bylo ponecháno.



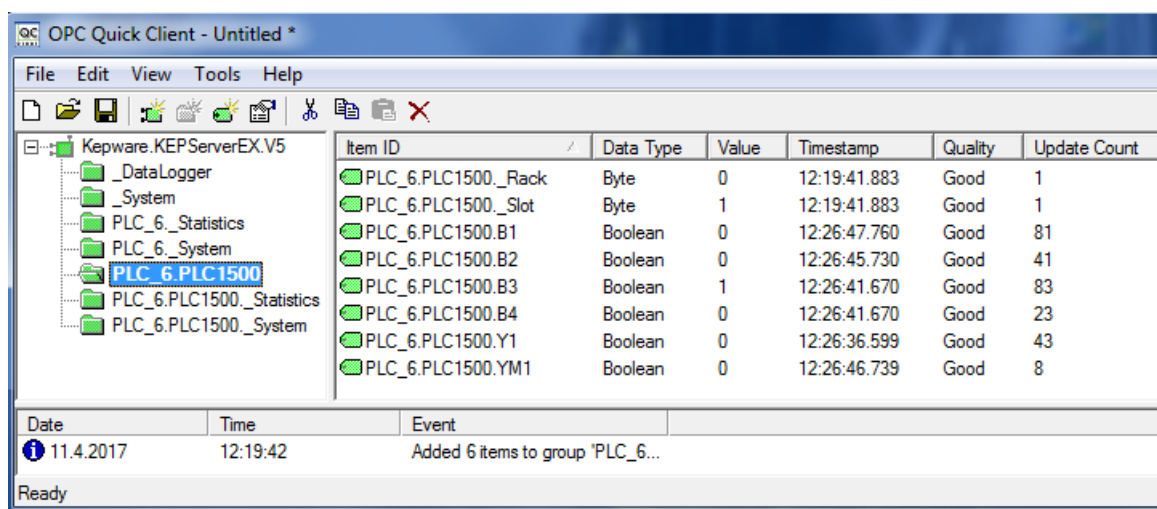
Obrázek 6.2: Přidání nového zařízení v KEPServerEX



Obrázek 6.3: Vkládání proměnných v KEPServerEX

Proměnné na OPC serveru jsou vytvořeny zadáním jejich jména, adresy a popisu. Adresu vstupních signálů je důležité zadávat jako memory, protože reálně do vstupních modulů PLC nejsou připojeny žádné fyzické vstupy. Adresu vstupních signálů je nutné přepsat i v TIA Portalu. Datový typ je vybrán automaticky podle zadané adresy. Pro klienta je možno ponechat přístup umožňující čtení i zápis.

Po vložení všech potřebných proměnných, lze pro kontrolu spojení spustit OPC Quick Client. V případě správné komunikace je zobrazena u proměnných kvalita „Good“ a je možné sledovat jejich aktuální hodnotu.



Obrázek 6.4: OPC Quick Client

Pro jednodušší zápis proměnných do OPC klienta lze v KEPServerEX vytvořit alias. V této bakalářské práci byl použit alias PLC nahrazující název PLC_6.PLC1500.

6.2 Nastavení OPC klienta

Pro správnou konfiguraci OPC klienta v SIMIT je nutné přidat stejnojmenné propojení. Do tohoto Coupling jsou pak zapsány všechny potřebné proměnné s využitím alias PLC a zvolíme typ proměnné. V případě analogových hodnot je zvolen datový typ integer, který umožňuje posílat hodnoty v rozsahu 0-27648. Multiplier udává četnost čtení nebo zápisu hodnoty proměnné na OPC server za n cyklů, přičemž n je nastavená hodnota.

U property Time slice je nastavena doba jednoho cyklu výměny dat, přičemž hodnota 2 odpovídá 100ms. Host name je ponecháno defaultně na localhost, v případě že se OPC server nachází na stejném PC. Pokud se OPC server nachází na jiném PC, je zde jeho IP adresa. Pro ProgID vybereme z nabídky vytvořený KEPServerEX.

The screenshot shows the 'OPC client (OPCClient)' configuration window. It has a 'Browse' button at the top. Below it are two sections: 'Inputs' and 'Outputs', each with a 'Reset filter' button. The 'Inputs' section contains a table with columns: Default, Name, Type, Multiplier, and Comment. The 'Outputs' section contains a table with columns: Name, Type, Multiplier, and Comment. At the bottom is a 'Properties' section with a table showing various settings.

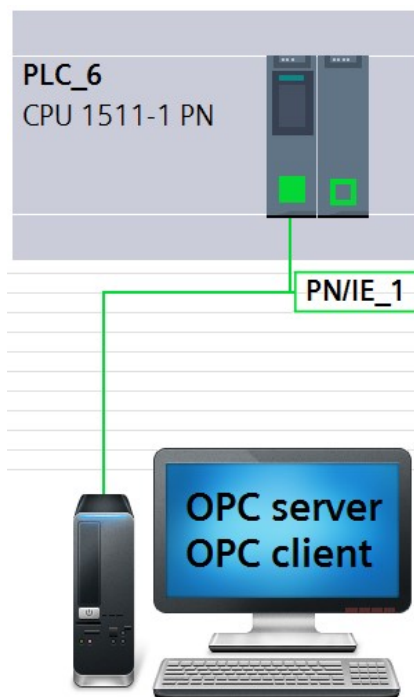
Inputs				
Default	Name	Type	Multiplier	Comment
<input type="checkbox"/>	PLC.B1	binary	1	Indukční snímač rotace pásu
<input type="checkbox"/>	PLC.B2	binary	1	Snímač přítomnosti puku
<input type="checkbox"/>	PLC.B3	binary	1	Vyhazovac zasunut
<input type="checkbox"/>	PLC.B4	binary	1	Vyhazovač vysunut

Outputs			
Name	Type	Multiplier	Comment
PLC.Y1	binary	1	Aktivace podavače puku
PLC.YM1	binary	1	Motor dopravníkového pásu

OPC client	
Property	Value
Time slice	2
Host name	localhost
ProgID	Kepware.KEPServerEX.V5
Status display	is_active

Obrázek 6.5: Nastavení OPC klienta v SIMIT

6.3 Testování aplikace na simulačním modelu



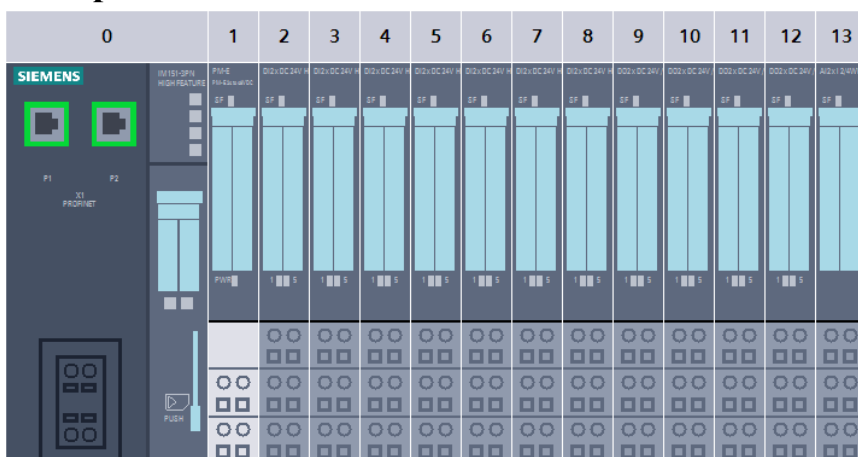
Obrázek 6.6: *Propojení PLC se SIMIT*

Simulační modely byly testovány již během svého vývoje, a tudíž šlo o otestování jednotlivých řídicích aplikací pro tyto modely. Po správné konfiguraci a vložení potřebných proměnných na OPC server a OPC klienta proběhlo ověření správnosti programu. Aplikace pro distribuční, testovací, procesní a třídící pracoviště byly vytvořeny bezchybně, a tudíž nebylo potřeba program ani vytvořený model nikterak modifikovat.

Řídicí program pro manipulační pracoviště obsahoval chyby, při kterých nedocházelo k přemístění servopohonu, vzhledem nedostatečně dlouhému impulzu na výstup Start-bit, a k okamžitému spouštění sekvence příkazů pro uvolnění puku z manipulační jednotky po projetí testovací pozice. První problém byl vyřešen za pomoci časovače TP, který umožnil vygenerovat dostatečně dlouhý impulz pro tento výstup. Druhá chyba byla vyřešena pomocí časovače TON, kdy musel servopohon setrvat v klidu alespoň 1 vteřinu a poté začala probíhat sekvence pro položení puku na jednu z výstupních pozic.

Aplikace mixovacího pracoviště byla udělána téměř správně, jedinou chybou zde bylo opačné zvolení matematického operátoru, čímž nedocházelo k postupnému vypouštění a čerpání vody zpět do zásobníků. Po úpravě všech networků s touto chybou, aplikace probíhala v pořádku.

6.4 Testování aplikace na laboratorním modelu



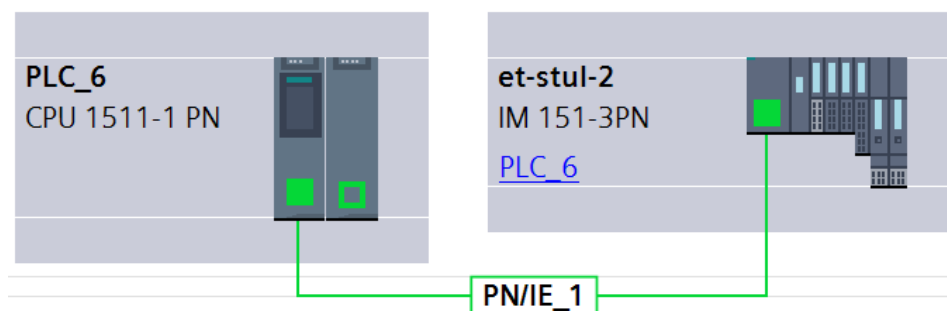
Obrázek 6.7: Hardware konfigurace modulu vzdálených vstupů/výstupů

Pro ověření správnosti aplikace bylo provedeno testování také na laboratorním modelu. Nezbytnou součástí bylo přidat do hardware konfigurace modul vzdálených vstupů a výstupů, který bylo potřeba také propojit s PLC v network view. Testování bylo provedeno na testovacím pracovišti.

Tabulka 6.1: Hardware konfigurace modulu vzdálených vstupů/výstupů a adresy

Slot	Modul	Číslo výrobku	Firmware	Vstupní adresy	Výstupní adresy
0	IM 151-3 PN	6ES7 151-3BA23-0AB0	V7.0		
1	PM-E 24 to 48VDC	6ES7 138-4CA50-0AB0			
2..8	2DI x 24VDC HF	6ES7 131-4BB01-0AB0		20..26	
9..12	2DO x 24VDC/0.5A HF	6ES7 132-4BB01-0AB0			20..23
13	2AI x I 2/4WIRE HF	6ES7 134-4MB02-0AB0		800..803	

Aby mohl být laboratorní model řízen PLC, bylo důležité také přepsat jednotlivé vstupní a výstupní adresy odpovídající adresám modulu vzdálených vstupů a výstupů. Nahráním aplikace a následným spuštěním PLC proběhl test, při kterém bylo zjištěno, že při vysunutí pístu pro měření výšky je puk pístem přitlačen na pás, čímž by mohlo dojít k poškození pásu. Při návrhu simulačního modelu nebylo s touto situací počítáno. Jednoduchou úpravou programu zajišťující vypnutí elektromotoru dopravníkového pásu, následné vysunutí a zasunutí pístu pro měření výšky a opětovné zapnutí dopravníku byla tato závada odstraněna.



Obrázek 6.8: Propojení PLC s modulem vzdálených vstupů/výstupů

Závěr

Přečtením vybrané literatury byly získány znalosti týkající se simulace systémů. Byly zjištěny a pochopeny nejčastěji používané přístupy v průmyslové automatizaci. Rozebráním několika simulačních nástrojů jsem získal přehled, jak detailně lze simulaci provádět. Následnou volbou simulačního prostředí SIMIT bylo úkolem se více seznámit s jednotlivými částmi. Prostudováním potřebných listů manuálu a vytvářením jednoduchých modelů jsem se naučil pracovat s tímto softwarem. Během programování na řídicím systému Simatic ve vývojovém prostředí TIA Portal jsem využil získané znalosti o programovatelných automatech během výuky na vysoké škole. Jednotlivé řídicí aplikace byly naprogramovány pomocí návrhu Petriho sítí s pomocnými stavovými paměťovými bity zajišťující sekvenční probíhání kódu.

Komunikace mezi PLC a simulačním modelem v software SIMIT by byla nejlépe provedena přes komunikační sběrnice PROFIBUS nebo PROFINET, které by umožnili rychlejší výměnu dat. K využití tohoto způsobu komunikace je zapotřebí interface modul, který by byl pro naše modely už zbytečný nadstandard. Ke komunikaci postačilo vytvořit OPC server v programu KEPServerEX, přes který probíhala výměna dat s menší odezvou, ale za to s minimálními náklady.

Během vývoje simulačních modelů bylo prováděno i jejich testování, tím že byly výstupy řízeny ručně v SIMIT. Tímto byly odhaleny jisté nedostatky předem a zavčasu. V průběhu testování bylo zjištěno, že řídicí aplikace pro distribuční, testovací, procesní i třídící pracoviště byly vytvořeny bezchybně a nebylo potřeba kód nijak upravovat. U zbývajících dvou pracovišť jsem se nevyhnul lehké úpravě kódu pro správné řízení modelu. Testování dále proběhlo i na laboratorním modelu vybraného pracoviště, kde bylo zapotřebí po prvním testu program jednoduše modifikovat.

Výsledkem bych shrnul, že využití simulace je podle mého názoru velmi přínosné. Na hotovém simulačním modelu je vývoj aplikace určitě rychlejší a pohodlnější. Software SIMIT bych zhodnotil jako skvělý nástroj pro vytváření simulačních modelů. Po získání všech potřebných znalostí je tvorba modelu v jeho intuitivním prostředí rychlá a jednoduchá. Jako nevýhodu bych u SIMIT považoval nedokončenou nápovědu, kdy nejsou některé komponenty dobře popsány a je nutno je vyhledávat v manuálu. Přidáním některých možností pro vizualizaci by podle mého názoru také nebylo na škodu.

Literatura

- [1] BANDYOPADHYAY, Susmita a BHATTACHARYA, Ranjan. *Discrete and Continuous Simulation: Theory and Practise*. Boca Raton, FL : CRC Press, 2014. ISBN: 978-1-4665-9639-9.
- [2] BANKS, Jerry, a další. *Discrete-Event System Simulation*. Upper Saddle River, NJ : Prentice Hall, 2004. ISBN: 0131446797.
- [3] SMITH, Roger D. Simulation: The Engine Behind the Virtual World. *Model Benders*. [Online] Prosinec 1999. [Citace: 25. Listopad 2016.] <http://www.modelbenders.com/papers/sim2000/SimulationEngine.PDF>.
- [4] SCHLAGER, Martin. *Hardware-in-the-Loop Simulation : A Scalable, Component-based, Time-triggered Hardware-in-the-loop Simulation Framework*. Saarbrücken : VDM Verlag Dr. Müller GmbH & Co. KG, 2011. ISBN: 978-3-8364-6216-7.
- [5] MOUGY, Amr El. NETW 707 Modeling and Simulation Amr El Mougy. *SlidePlayer*. [Online] [Citace: 3. Leden 2017.] <http://slideplayer.com/slide/4191779/>.
- [6] Using V Models for Testing . *SEI Insights*. [Online] 11. Listopad 2013. [Citace: 19. Prosinec 2016.] https://insights.sei.cmu.edu/sei_blog/2013/11/using-v-models-for-testing.html.
- [7] Model-Based Systems Engineering and Control System Development via Virtual Hardware-in-the-Loop Simulation. *Autonomie*. [Online] Říjen 2010. [Citace: 20. Prosinec 2016.] http://www.autonomie.net/docs/5%20-%20Presentations/CIL/modelbased_systems_engineering_ppt.pdf.
- [8] MIL simulation - Model in the loop. *Rexcontrols*. [Online] [Citace: 19. Prosinec 2016.] <https://www.rexcontrols.com/mil-simulation>.
- [9] SIL simulation - Software in the loop. *Rexcontrols*. [Online] [Citace: 19. Prosinec 2016.] <https://www.rexcontrols.com/sil-simulation>.
- [10] Model-based Design Definition of Terms. *Autonomie*. [Online] [Citace: 19. Prosinec 2016.] http://www.autonomie.net/%5Creferences/model_based_design_defs_24c.html.
- [11] PIL simulation - Processor in the loop. *Rexcontrols*. [Online] [Citace: 19. Prosinec 2016.] <https://www.rexcontrols.com/pil-simulation>.
- [12] HIL simulation - Hardware in the loop. *Rexcontrols*. [Online] [Citace: 19. Prosinec 2016.] <https://www.rexcontrols.com/hil-simulation>.
- [13] Oferta. *Technologie dla przemysłu - Taskoprojekt*. [Online] [Citace: 3. Leden 2017.] <http://taskoprojekt.com.pl/oferta,5.html>.
- [14] *Simulation of Manufacturing Systems*. LAW, Averill M. a McCOMAS, Michael G. Washington, D.C. : IEEE Computer Society Press, 1997. stránky 86-89. ISBN: 0-7803-5134-7.
- [15] Discrete Event, Continuous and Hybrid Dynamic Modeling. *GoldSim*. [Online] [Citace: 27. Prosinec 2016.] <http://www.goldsim.com/Web/Introduction/SimulationTypes/>.

- [16] Discrete Event - AnyLogic Simulation Software. *Multimethod Simulation Software and Solutions*. [Online] [Citace: 13. Leden 2017.] <http://www.anylogic.com/discrete-event-simulation>.
- [17] JAHANGIRIAN, Mohsen, a další. *Simulation in manufacturing and business: A review*. Londýn : European Journal of Operational Research, 12. Červen 2009.
- [18] RUBINSTEIN, Reuven Y. a KROESE, Dirk P. *Simulation and the Monte Carlo Method*. místo neznámé : Wiley-Interscience, 2008. ISBN: 978-0-470-17794-5.
- [19] SWEETSER, Al. *A Comparison of System Dynamics (SD) and Discrete Event Simulation (DES)*. Washington, DC : Andersen Consulting.
- [20] System Dynamics - AnyLogic Simulation Software. *Multimethod Simulation Software and Solutions*. [Online] [Citace: 13. Leden 2017.] <http://www.anylogic.com/system-dynamics>.
- [21] Agent Based Modeling - AnyLogic Simulation Software. *Multimethod Simulation Software and Solutions*. [Online] [Citace: 13. Leden 2017.] <http://www.anylogic.com/agent-based-modeling>.
- [22] MUSTAFEE, Navonil. *Operational Research for Emergency Planning in Healthcare: Volume 2*. místo neznámé : Palgrave Macmillan UK, 2016. ISBN: 978-1-137-57326-1.
- [23] KRACÍK, Lukáš. Jak využít simulace Monte Carlo ve financích. *Měsíc.cz - váš průvodce finančním světem*. [Online] 5. Červen 2015. [Citace: 3. Leden 2017.] <http://trhy.mesec.cz/clanky/jak-vyuzit-simulace-monte-carlo-ve-financich/>.
- [24] *Elements of a Hybrid Simulation Model: A Case Study of the Blood Supply Chain in Low and Middle Income Countries*. ONGGO, Bhakti Stephan. Lancaster : Lancaster University, 2014. Winter Simulation Conference.
- [25] Automated guided vehicle. *Wikipedia, the free encyclopedia*. [Online] [Citace: 4. Leden 2017.] https://en.wikipedia.org/wiki/Automated_guided_vehicle.
- [26] MIHAI, Dan. Visual Components: an Advanced Production Line Simulation Software. *Smashing Robotics*. [Online] 25. Leden 2016. [Citace: 5. Leden 2017.] <https://www.smashingrobotics.com/visual-components-production-line-simulation-software/>.
- [27] Visual Components 4.0 - The 3D factory simulation product family. *Factory simulation software - Visual Components*. [Online] [Citace: 5. Leden 2017.] <http://visualcomponents.com/visual-components-4-0/>.
- [28] Visual Components Essentials - Design the factories of the future. *Factory simulation software - Visual Components*. [Online] [Citace: 5. Leden 2017.] <http://www.visualcomponents.com/visual-components-4-0/essentials/>.
- [29] Visual Components Professional - The component creation solution. *Factory simulation software - Visual Component*. [Online] [Citace: 7. Leden 2017.] <http://www.visualcomponents.com/visual-components-4-0/professional/>.
- [30] Visual Components Premium - The factory simulation solution. *Factory simulation software - Visual Components*. [Online] [Citace: 7. Leden 2017.] <http://www.visualcomponents.com/visual-components-4-0/premium/>.

- [31] *SIMIT Simulation Software - SIMIT 9*. Karlsruhe : Siemens AG, 2016.
- [32] *SIMATIC SIMIT (V9.0) - Operating Manual*. Norimberk : Siemens AG, 2016. A5E38238767-AA.
- [33] COMOS Walkinside - Plant Engineering Software. *Siemens Global Website*. [Online] [Citace: 11. Leden 2017.] <http://w3.siemens.com/mcms/plant-engineering-software/en/comos-lifecycle/comos-walkinside/pages/default.aspx>.
- [34] Press Releases. *Siemens Global Website*. [Online] 20. Duben 2016. [Citace: 11. Leden 2017.] [http://www.siemens.com/press/en/pressrelease/?press=/en/pressrelease/2016/processindustries-drives/pr2016040239pden.htm&content\[\]=PD](http://www.siemens.com/press/en/pressrelease/?press=/en/pressrelease/2016/processindustries-drives/pr2016040239pden.htm&content[]=PD).
- [35] COMOS Walkinside Viewer - Plant Engineering Software. *Siemens Global website*. [Online] [Citace: 5. Leden 2017.] http://www.siemens.co.in/en/about_us/index/our_business_segments/industry/automation_and_drives/comoswalkinsideviewer.htm.
- [36] COMOS Walkinside Builder- Plant Engineering Software. *Siemens Global Website*. [Online] [Citace: 11. Leden 2017.] <http://w3.siemens.com/mcms/plant-engineering-software/en/comos-lifecycle/comos-walkinside/walkinside-builder/Pages/Default.aspx>.
- [37] COMOS Walkinside ITS (Immersive Training Simulator) - Plant Engineering Software. *Siemens Global Website*. [Online] [Citace: 11. Leden 2017.] <http://w3.siemens.com/mcms/plant-engineering-software/en/comos-lifecycle/comos-walkinside/walkinside-its/Pages/Default.aspx>.
- [38] About Us - AnyLogic Simulation Software. *Multimethod Simulation Software and Solutions*. [Online] [Citace: 12. Leden 2017.] <http://www.anylogic.com/about-us>.
- [39] Why AnyLogic? - AnyLogic Simulation Software. *Multimethod Simulation Software and Solutions*. [Online] [Citace: 12. Leden 2017.] <http://www.anylogic.com/features>.
- [40] Application Areas - AnyLogic Simulation Software. *Multimethod Simulation Software and Solutions*. [Online] [Citace: 12. Leden 2017.] <http://www.anylogic.com/application-areas>.
- [41] Editions Comparison - AnyLogic Simulation Software. *Multimethod Simulation Software and Solutions*. [Online] [Citace: 12. Leden 2017.] <http://www.anylogic.com/editions-comparison>.
- [42] Free Simulation Software for Education. *Multimethod Simulation Software and Solutions*. [Online] [Citace: 12. Leden 2017.] <http://www.anylogic.com/download-free-simulation-software-for-education/>.
- [43] Help - AnyLogic Simulation Software. *Multimethod Simulation Software and Solutions*. [Online] [Citace: 12. Leden 2017.] <http://help.anylogic.com/index.jsp?topic=/com.xj.anylogic.help/html>.
- [44] Devize Example - Automotive - Minitab. *Minitab*. [Online] [Citace: 7. Leden 2017.] <https://www.minitab.com/en-us/monte-carlo/automotive/>.
- [45] Company - Minitab. *Minitab*. [Online] [Citace: 8. Leden 2017.] <https://www.minitab.com/en-us/company/>.
- [46] Devize Workflow - Minitab. *Minitab*. [Online] [Citace: 7. Leden 2017.] <https://www.minitab.com/en-us/products/devize/workflow/>.

[47] Devize Reasults - Minitab. *Minitab*. [Online] [Citace: 7. Leden 2017.] <https://www.minitab.com/en-us/products/devize/results/>.

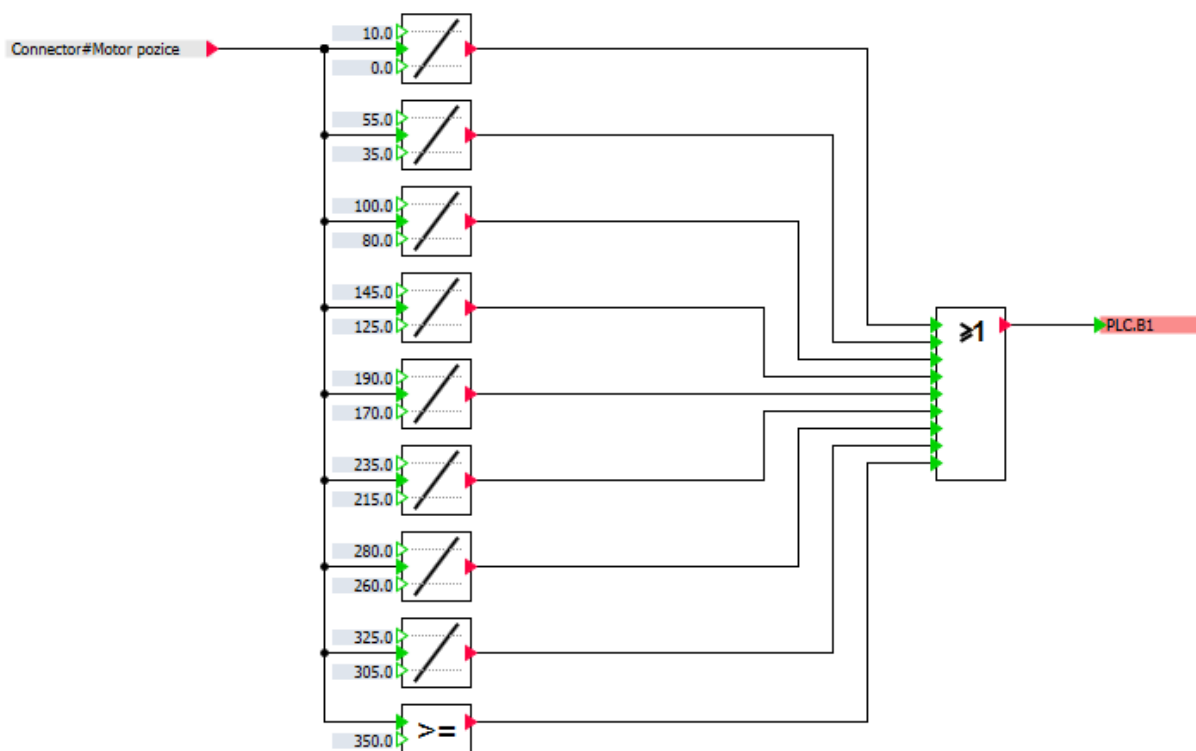
[48] Virtual Commissioning - ReplicAnt VR Demo - VR Process Simulation And 3D SCADA. *YouTube*. [Online] 26. Srpen 2014. [Citace: 13. Leden 2017.] <https://www.youtube.com/watch?v=xvyKlOdhRX4>.

[49] ReplicAnt VR. *ANT Automation Applying New Technologies*. [Online] [Citace: 13. Leden 2017.] <http://ant-automation.net/replicant-vr/>.

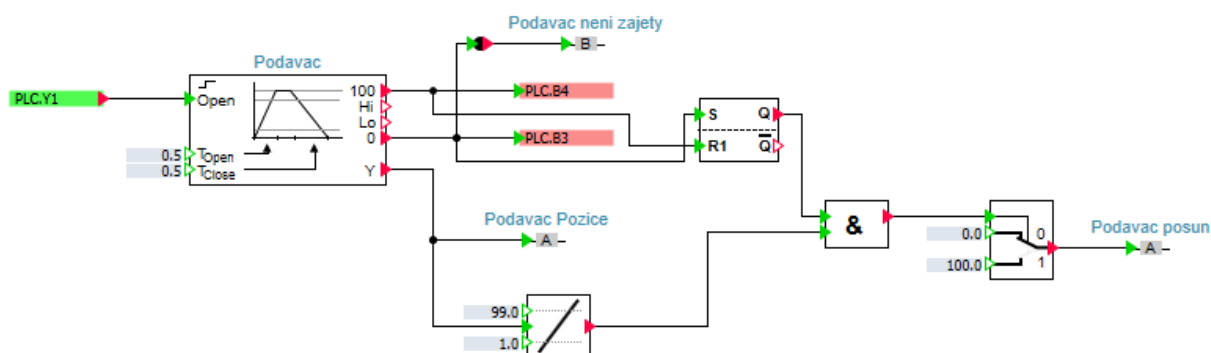
Seznam příloh

Příloha A:	<i>Diagramy distribučního pracoviště</i>	I
Příloha B:	<i>Diagramy testovacího pracoviště</i>	III
Příloha C:	<i>Diagramy procesního pracoviště</i>	V
Příloha D:	<i>Diagramy manipulačního pracoviště</i>	IX
Příloha E:	<i>Diagramy třídícího pracoviště</i>	XIII
Příloha F:	<i>Diagramy mixovacího pracoviště</i>	XV
Příloha G:	<i>CD-ROM</i>	XVI

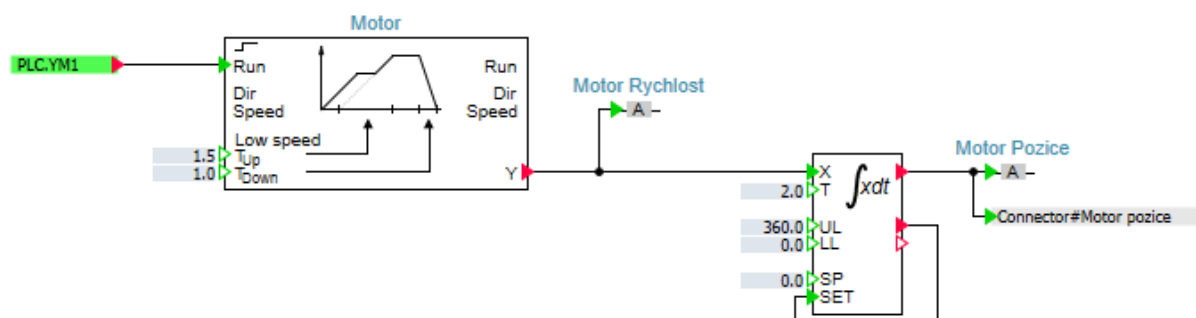
Příloha A: *Diagramy distribučního pracoviště*



Obrázek A.1: *Snímač rotace pásu*



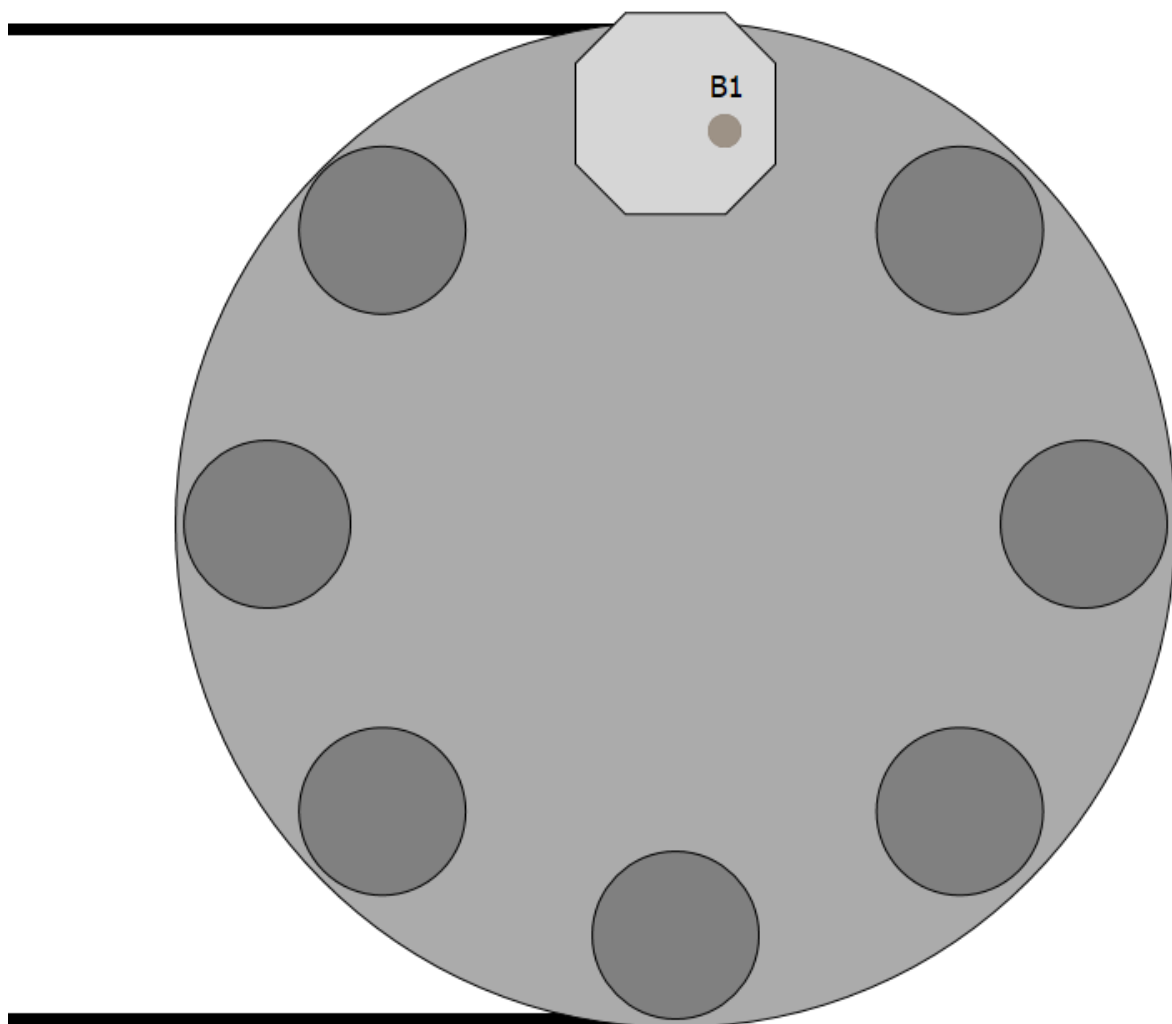
Obrázek A.2: *Podavač puku*



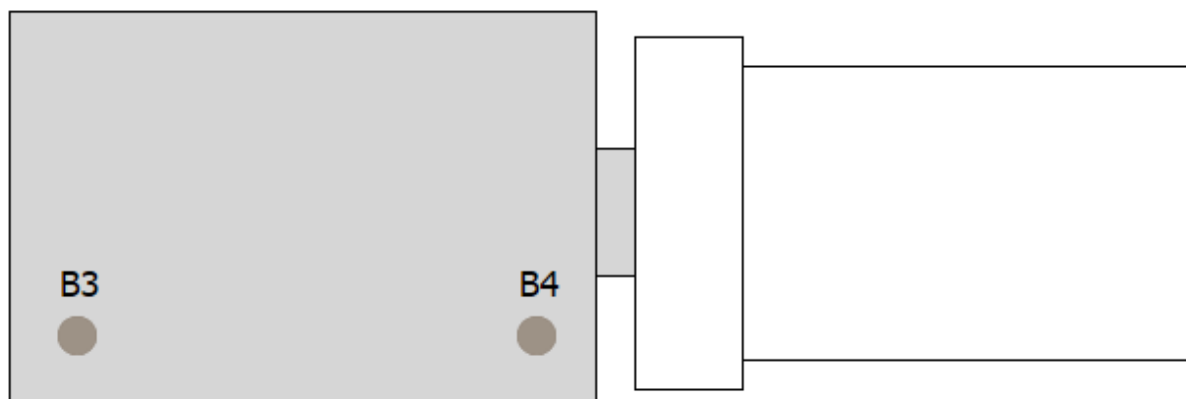
Obrázek A.3: *Motor pásu*



Obrázek A.4: *Snímač přítomnosti*

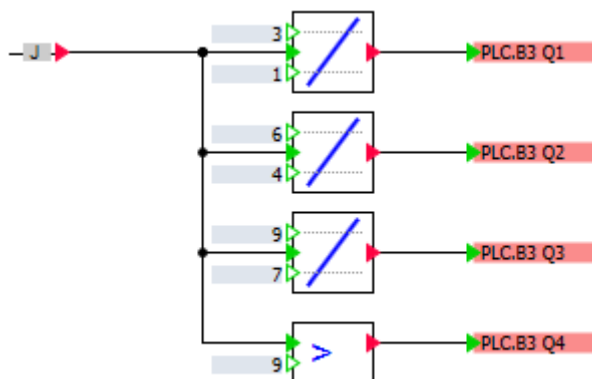


Obrázek A.5: *Vizualizace snímače rotace pásu*

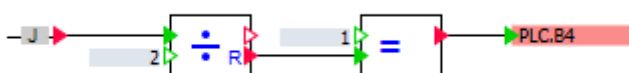


Obrázek A.6: *Vizualizace podavače puku*

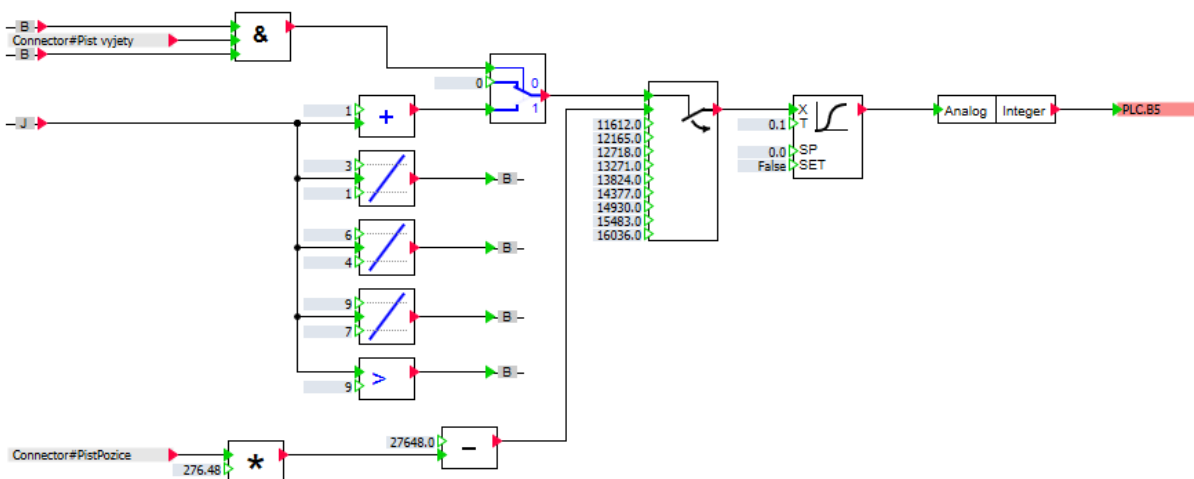
Příloha B: *Diagramy testovacího pracoviště*



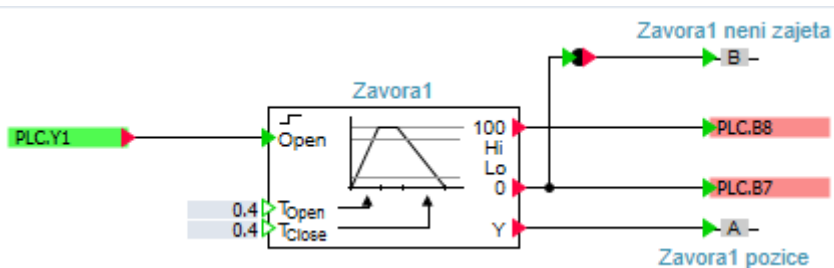
Obrázek B.1: *Snímač barev*



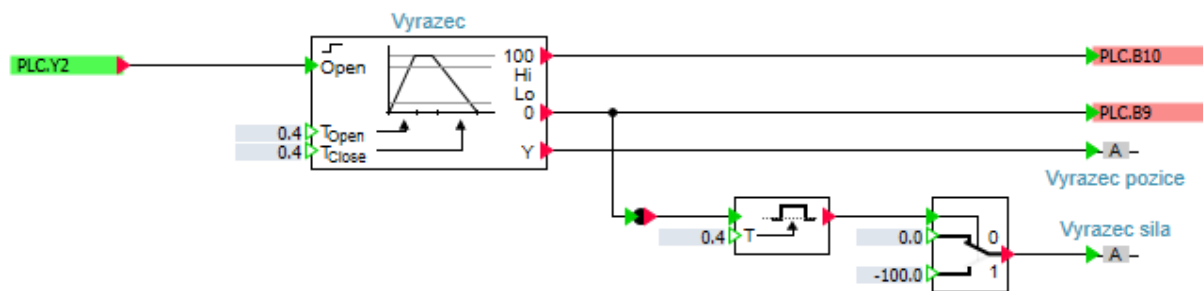
Obrázek B.2: *Snímač magnetických vlastností*



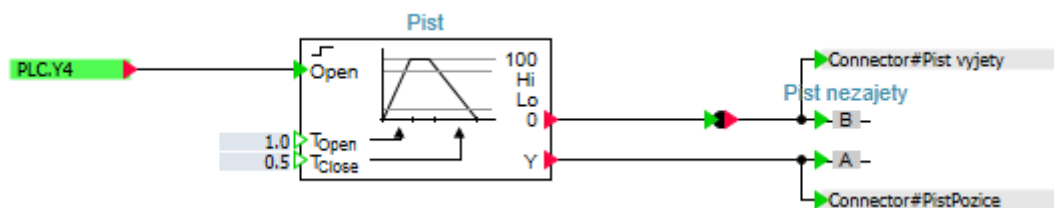
Obrázek B.3: *Senzor výšky*



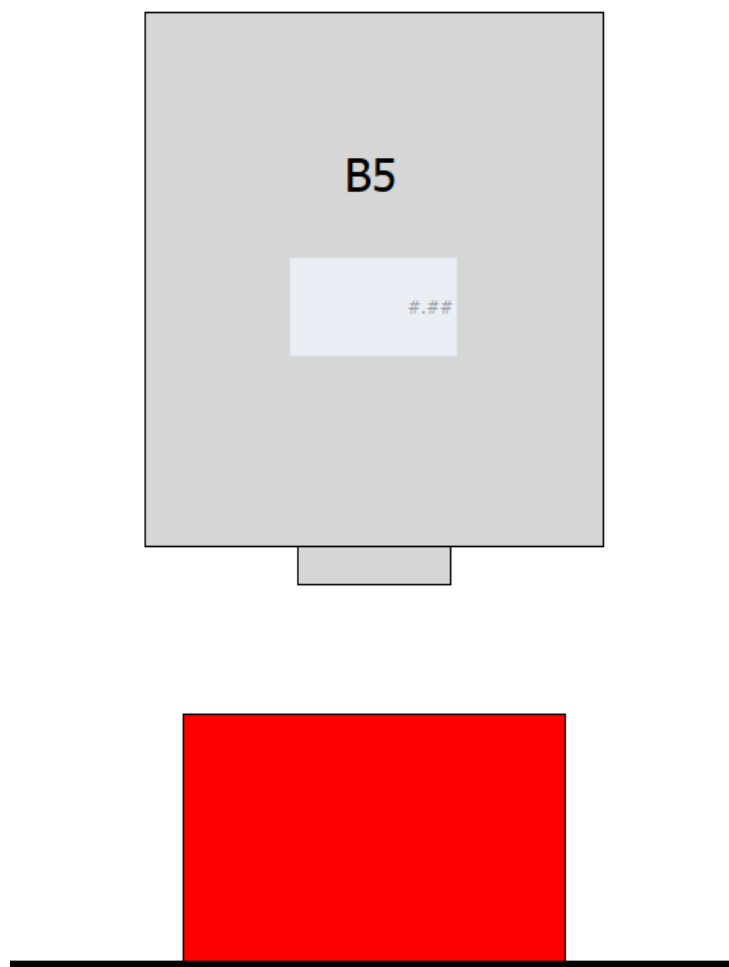
Obrázek B.4: *Závora č.1*



Obrázek B.5: *Vyrážec*



Obrázek B.6: *Píst pro měření výšky*



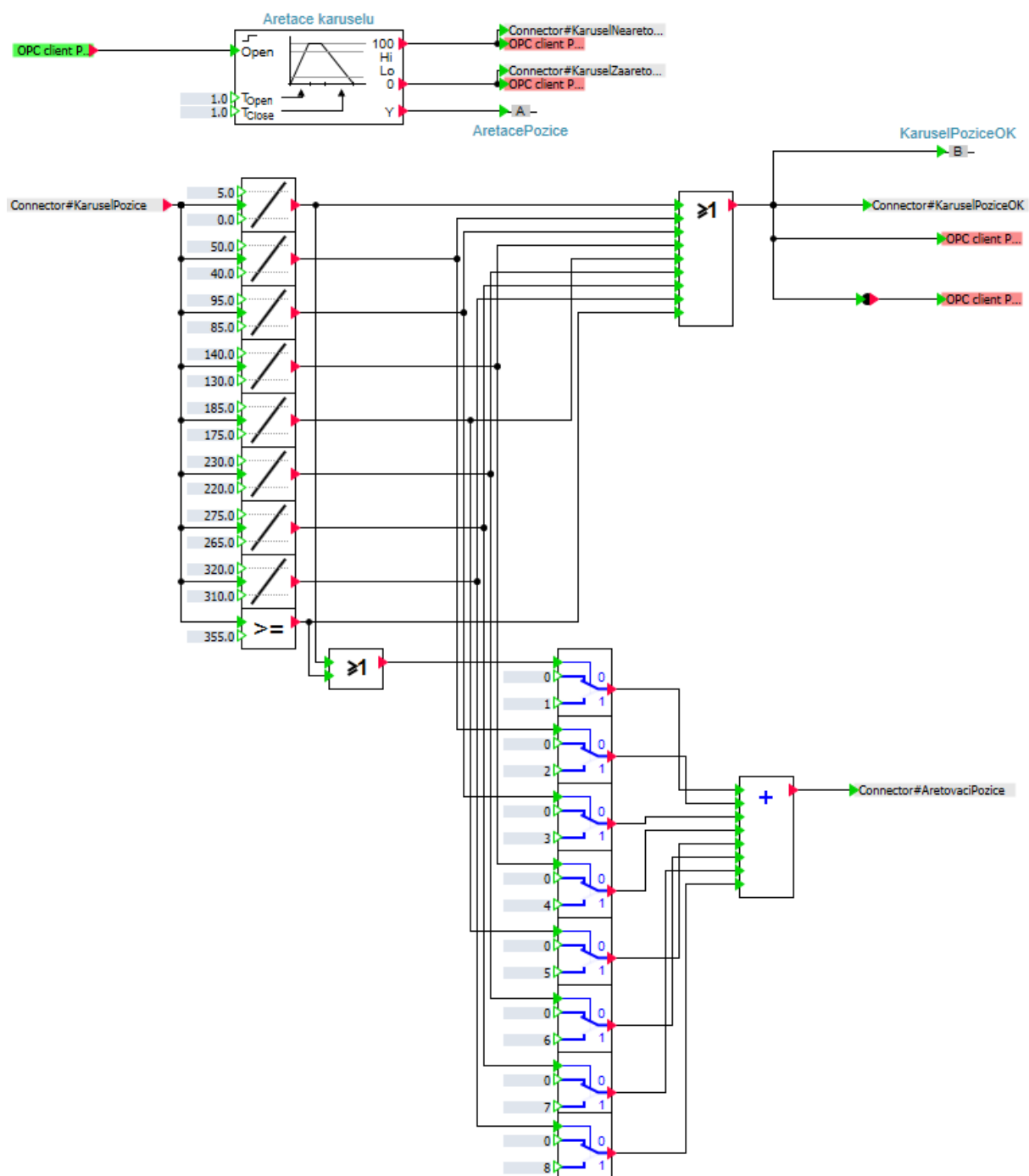
Obrázek B.7: *Vizualizace senzoru výšky*

Příloha C: *Diagramy procesního pracoviště*

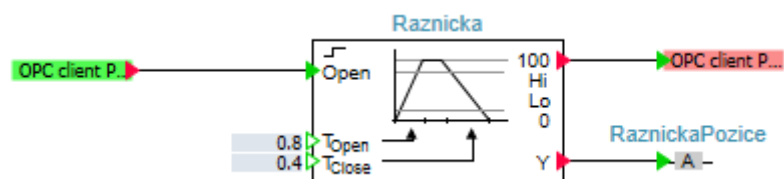
PasSensorB1



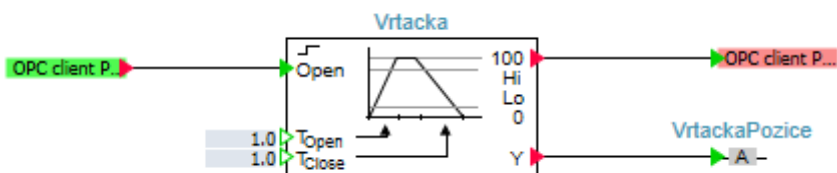
Obrázek C.1: *Puk pod chapadly*



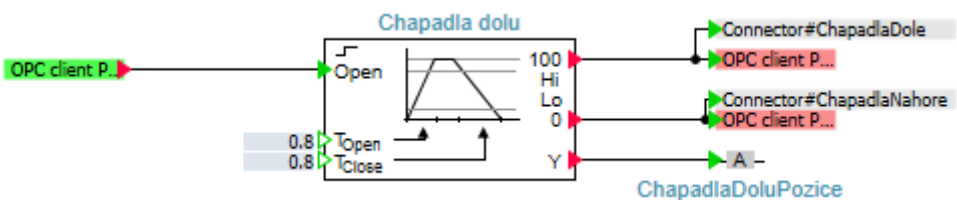
Obrázek C.2: *Aretace karuselu*



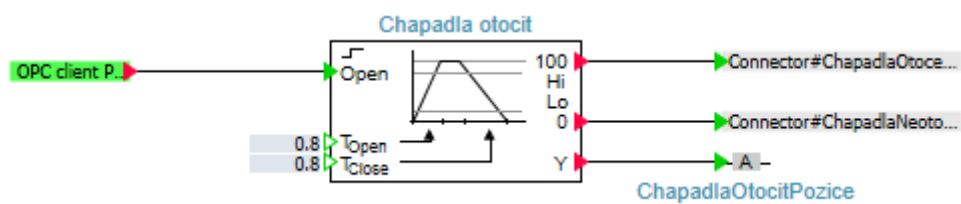
Obrázek C.3: *Raznicka*



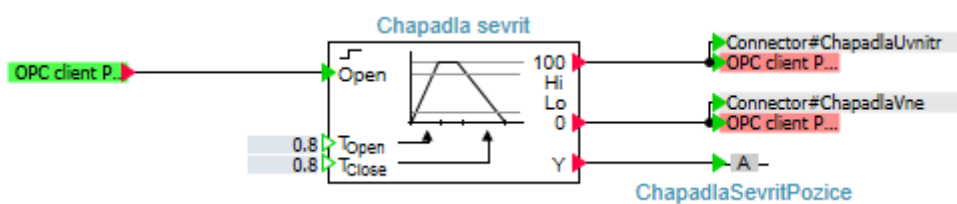
Obrázek C.4: *Vrtačka*



Obrázek C.5: *Chapadla dolu*



Obrázek C.6: *Chapadla otočit*

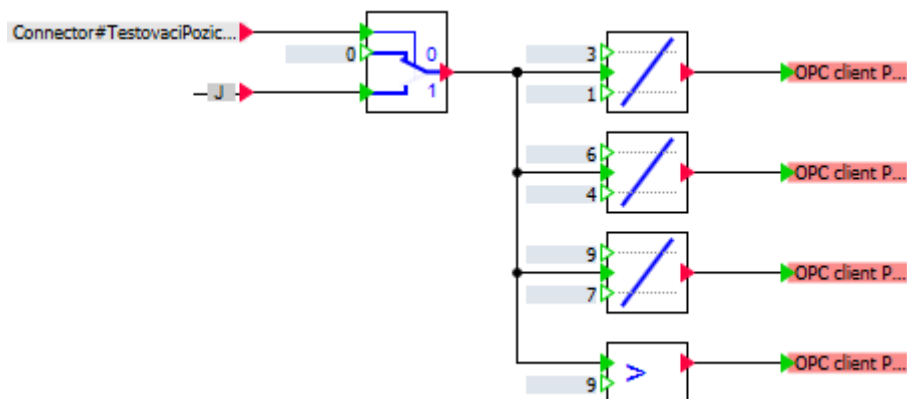


Obrázek C.7: *Chapadla sevřít*

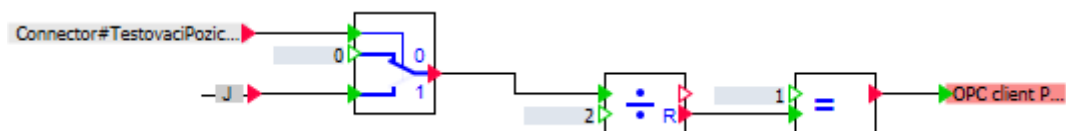


Obrázek C.8: *Ofuk*

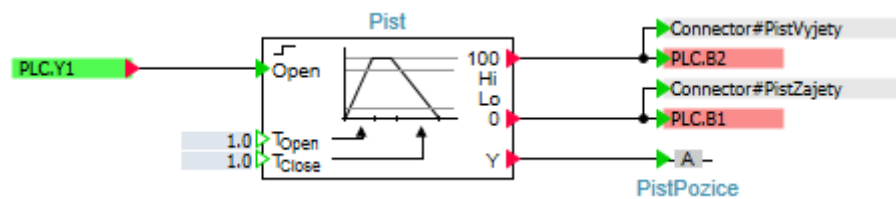
Příloha D: *Diagramy manipulačního pracoviště*



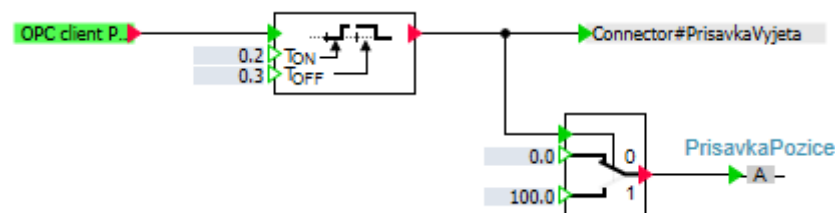
Obrázek D.1: *Snímač barev*



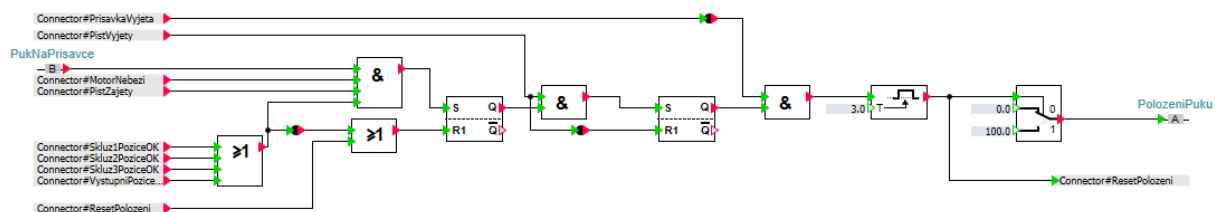
Obrázek D.2: *Snímač magnetických vlastností*



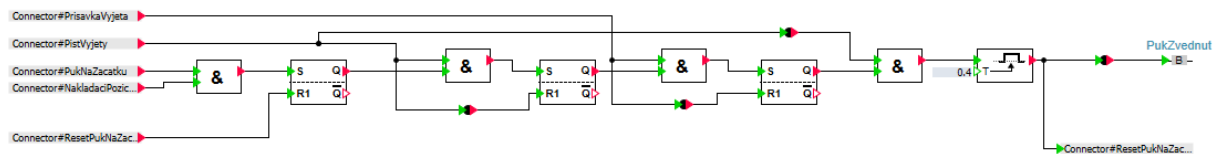
Obrázek D.3: *Píst s přísavkou*



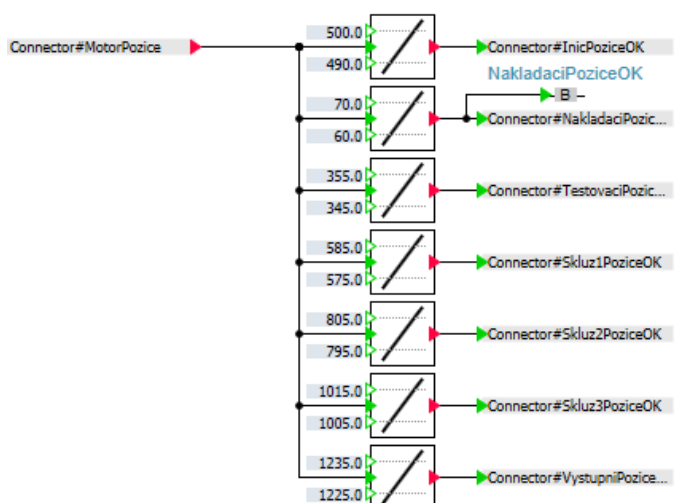
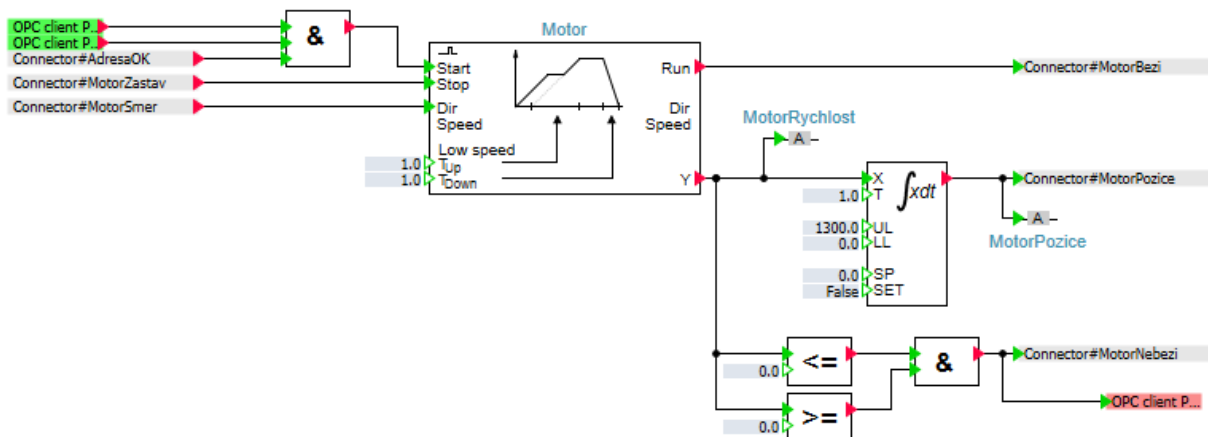
Obrázek D.4: *Přísavka*



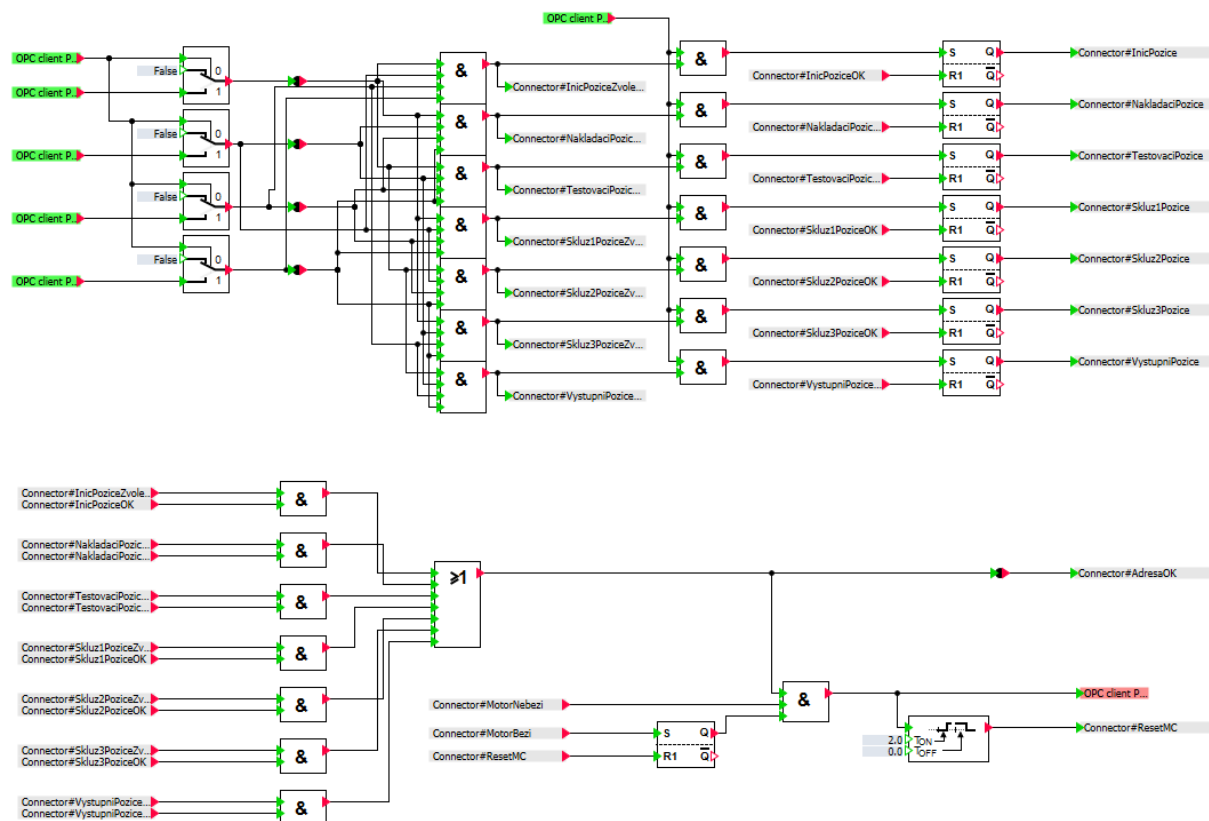
Obrázek D.5: *Položení puku*



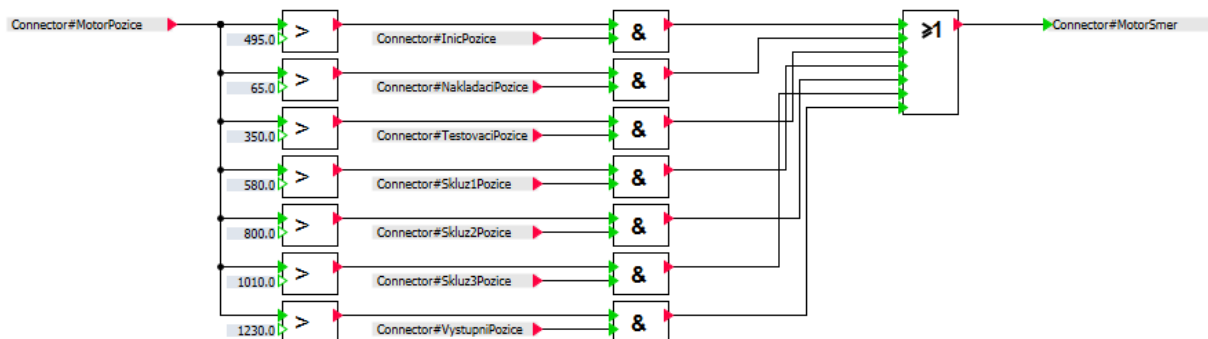
Obrázek D.6: *Zdvih puku*



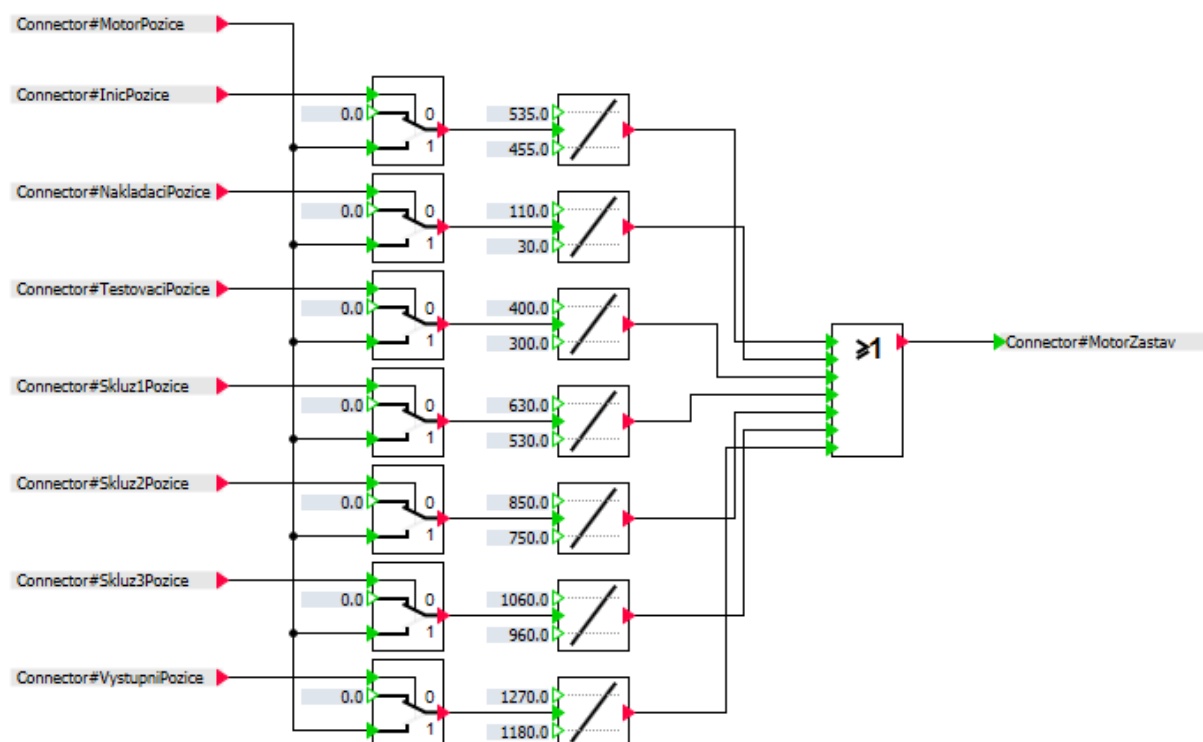
Obrázek D.7: *Motor servopohonu*



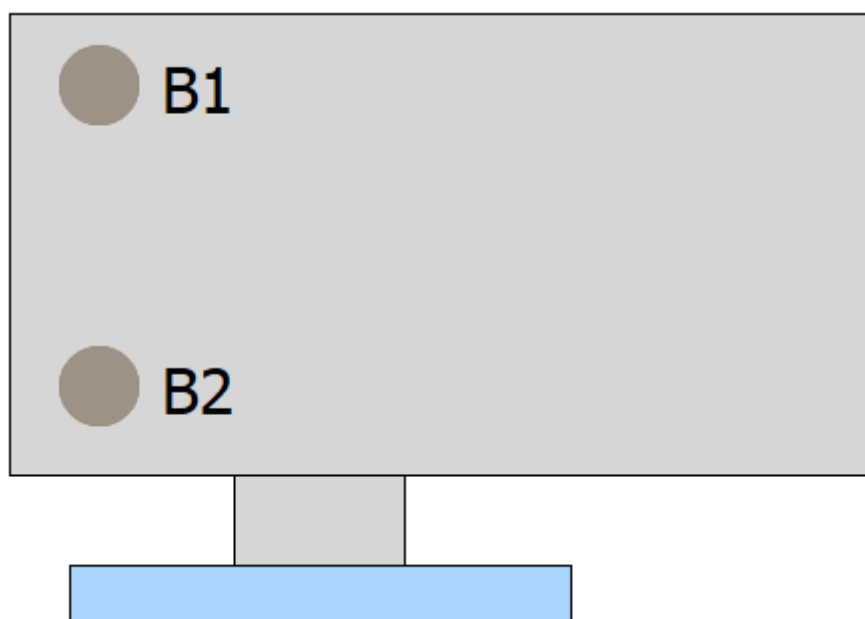
Obrázek D.8: Přesun servopohonu



Obrázek D.9: Směr motoru servopohonu

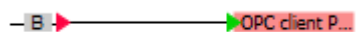


Obrázek D.10: Zastavení motoru servopohonu

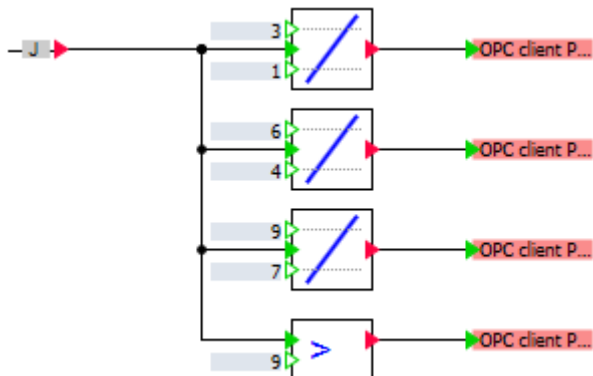


Obrázek D.11: Vizualizace manipulační jednotky

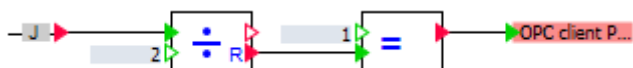
Příloha E: *Diagramy třídícího pracoviště*



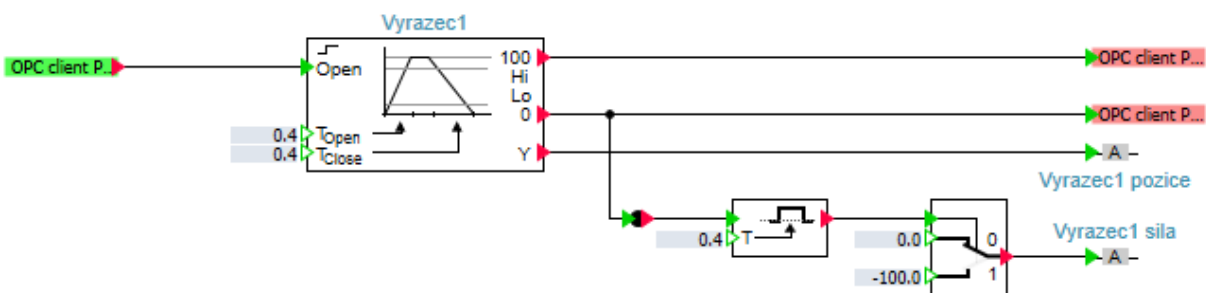
Obrázek E.1: *Snímač přítomnosti č. 1*



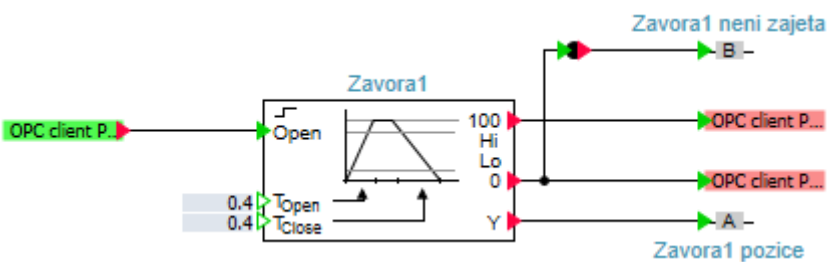
Obrázek E.2: *Snímač barev*



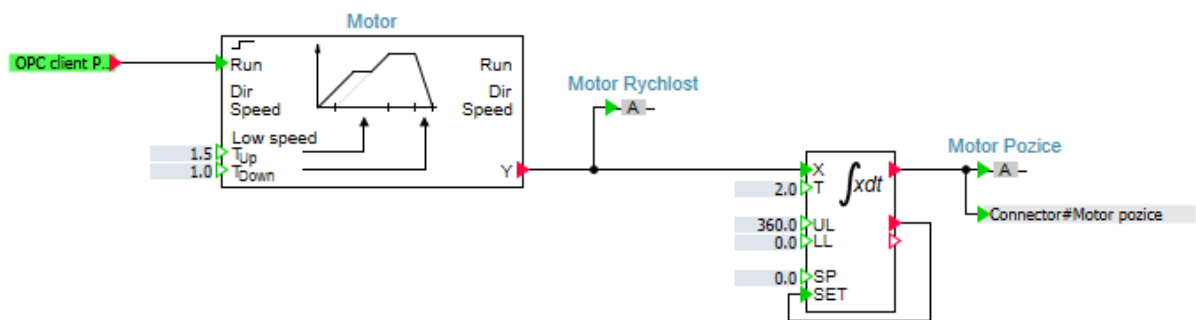
Obrázek E.3: *Snímač magnetických vlastností*



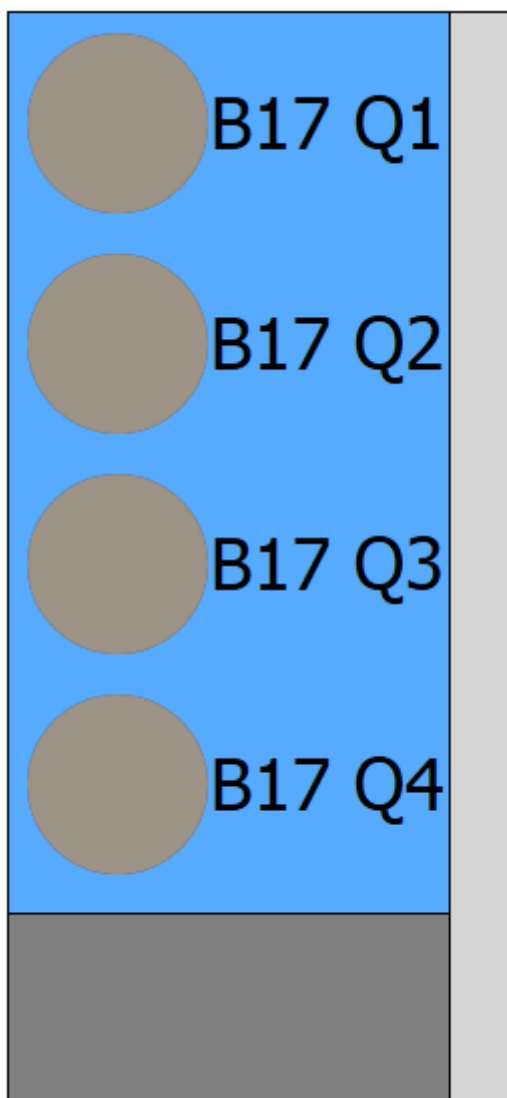
Obrázek E.4: *Vyrážeč č. 1*



Obrázek E.5: *Závora č. 1*

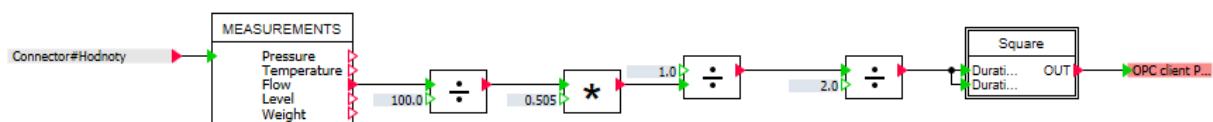


Obrázek E.6: *Motor pásu*

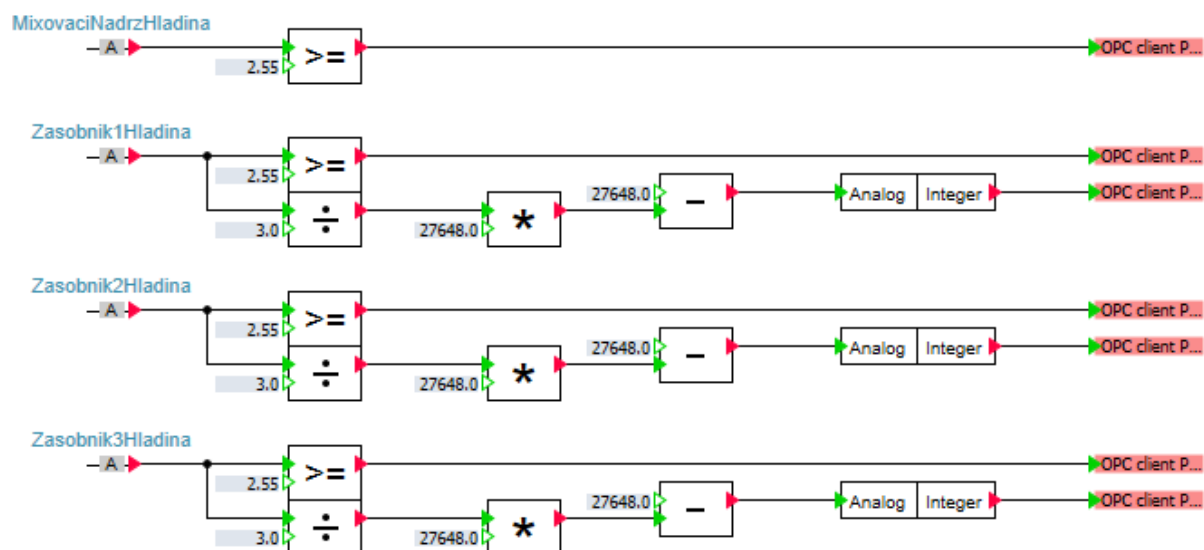


Obrázek E.7: *Vizualizace snímače barev*

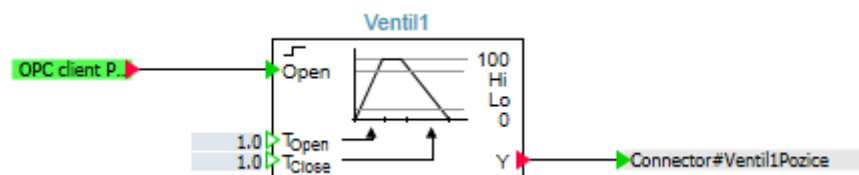
Příloha F: *Diagramy mixovacího pracoviště*



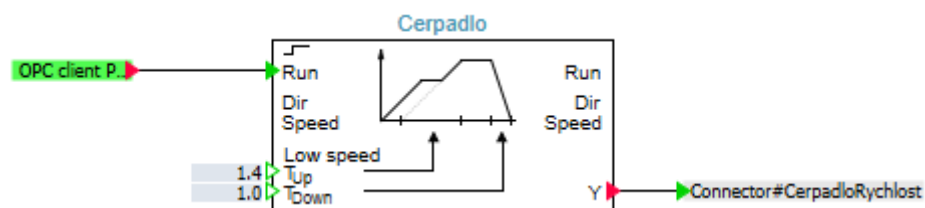
Obrázek F.1: *Průtokoměr*



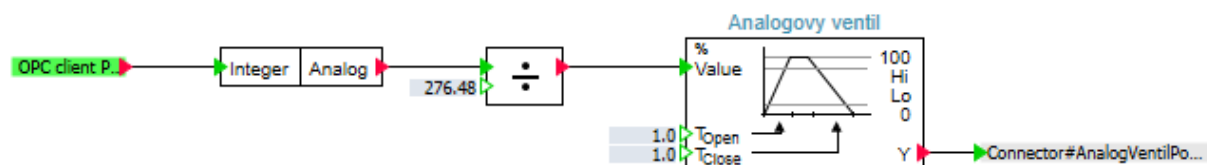
Obrázek F.2: *Snímače výšky*



Obrázek F.3: *Ventil č. 1*



Obrázek F.4: *Čerpadlo*



Obrázek F.5: *Analogový ventil*

Příloha G: *CD-ROM*

- Elektronická verze bakalářské práce ve formátu PDF
- Simulační modely pro distribuční, testovací, procesní, manipulační, třídící a mixovací pracoviště vytvořené v SIMIT 9
- Řídící aplikace pro distribuční, testovací, procesní, manipulační, třídící a mixovací pracoviště vytvořené v TIA Portal V13
- Nastavení OPC Serveru pro distribuční, testovací, procesní, manipulační, třídící a mixovací pracoviště vytvořené v KEPServerEX 5.0
- Manuál k SIMIT 9